Waseda University Doctoral Dissertation

Controlling Value Based Pseudo Power Gating for Power-efficient LSI Design

Yu JIN

Graduate School of Information, Production and Systems
Waseda University
February 2014

Abstract

Continuous increase in chip density and switching speed of LSI circuits enables over 100 million of integrated gates running at several Giga-hertz in a chip. However, high density and switching speed result in significant power consumption and the demand for low power LSI design has been accelerated significantly. Low power design methodologies have been studied at all design levels for reducing dynamic power and static power. Dynamic power relates to the output changes of logic gates and static power relates to the leak current of them. Dynamic power reduction is important to reduce the runtime power for high performance operations.

The dynamic power dissipation is affected by switching activity of each gate and an avoidance of unnecessary switching activities is effective to reduce the dynamic power. To avoid unnecessary switching activities, a controlling value (CV) of a logic gate takes an important role. Focusing on one gate, if one input takes a CV, then switching of other inputs do not affect to the output of the gate. The CV can stop unnecessary output changes of the focused gate, but there exist unnecessary changes in gate blocks computing other inputs. In other words, switching activities in blocks computing other inputs are unnecessary and give the waste of the dynamic power when one input takes CV. In entire circuit, such power dissipation accounts for a significant amount of overall power consumption. Thus an effective method to reduce redundant switching activities is expected for improving the power efficiency.

In this dissertation, Pseudo Power Gating (Pseudo PG) method based on CV of gates is proposed to save such waste of power for computing unnecessary input values. In Pseudo PG, an input taking the CV of one gate is added as a control signal to gate blocks for other inputs to stop unnecessary changes. The control signal is used to stop the output change of each gate in the block for reducing dynamic power. Distinguished from other power gating

methods, the proposed method can reduce dynamic power efficiently by optimizing the switching activities in ultra-fine grained region without extra hardware circuitry. Several heuristic block decision algorithms are proposed for applying Pseudo PG method on combinational circuits. In addition, Pseudo PG is enhanced to dual-stage structure for further reduction of potential power dissipation.

This dissertation consists of 6 Chapters organized as follows:

Chapter 1 [Introduction] briefly summarizes current low power technologies for different levels of LSI design including the gate level circuits. The related works of reducing dynamic and static power are introduced and their advantages and drawbacks are clarified and several challenges for saving dynamic dissipation are observed. The objective and scope of this research are indicated and the motivations and contributions of this dissertation are described.

Chapter 2 [Preliminaries] shows controlling value based power gating method for reducing static power as a basis of our proposal. Binary Decision Diagram (BDD) is also introduced. BDD is a data structure to represent a Boolean function and logic operations can be done on BDD's. The method to estimate switching activities by using BDD is also presented in this chapter for optimizing the dynamic power.

Chapter 3 [Pseudo Power Gating Method based on SW-first Algorithm] proposes a CV based Pseudo PG method to optimize the power consumption of combinational circuits by reducing the switching activities. CV of a gate can inactivate other inputs and the switching activities of the blocks computing other inputs can be reduced by the input taking CV as a control signal.

In Pseudo PG, a control signal is associated to the controlled gates. In other words, a control signal is related to a cluster of gates. The clustering algorithm to identify the optimum control signal and corresponding

power-controlled blocks is essential. A basic size based algorithm and a switching activity first (*SW*-first) algorithm are proposed. In the basic algorithm, the candidate control signals are sorted with the number of controllable gates, one with the maximum number is selected as a control signal, and so on. On the other hand, the *SW*-first algorithm manipulates the reduction of switching activates by adding a control signal, and maximizes the total reduction. Control signals are sorted by the reduction, and selected a signal with the higher reduction one by one.

The proposed method has been implemented in C language and applied to ISCAS85 standard benchmark circuits based on the existing BDD manipulation system. Experimental results show that 0.93%~11.56% of reduction of switching activity is obtained by the basic size based algorithm and 1.85%~35.12% reduction is obtained by the SW-first algorithm. The depth increase after inserting a control signal is considered and the steady maximum depth constraint is applied to prevent performance degradation. Afterward, the power consumption is evaluated by simulating the netlist which is extracted from the layout using the VDEC (VLSI Design and Education Center in Tokyo University) Rohm 0.18μm process. Experiments on the ISCAS'85 benchmarks show that the reduction of total power consumption has been achieved to 2.99%-25.12% (12.77% on average). This work provides a solution for switching activity reduction in very fine-grained region of logic level LSI design with small overhead.

Chapter 4 [Transitive Fan-in based Algorithm for Further Power Saving] propose a transitive fan-in based algorithm to obtain a control signal with smaller depth under the steady maximum depth constraint. By the constraint, several gates are excluded from a power-controlled block, and considerable potential power reduction is lost. We found that a CV might be propagated from a gate to another gate on a path of serially connected gates. The proposed transitive fan-in based algorithm finds control signal candidates with smaller depth based on the propagation of controlling condition via a fan-in path, with which more

opportunities can be obtained.

As a result, power reduction achieved by the transitive fan-in based algorithm ranges from 3.62% to 27.72% (19.05% on average) and exceeds the previous SW-first algorithm by 6.28%. The transitive fan-in based algorithm have obvious advantages in detection of control condition favorable for power reduction.

Chapter 5 [Dual-Stage Pseudo Power Gating] newly introduces dual-stage Pseudo PG where up to two extra control signals are added to a gate if effective. No matter how effective clustering algorithms are used, there still exist unnecessary computations in power-controlled blocks when their control signals take non-controlling value. The dual-stage structure with two control signals improves power reduction capability. When one signal might take a non-controlling value, another signal can stop the activity. Dual-stage structure is realized by adopting another Pseudo PG to the inside of a power-controlled block to stop unnecessary computation while the single-stage control signal takes non-effective value.

The proposed dual-stage Pseudo PG structure is implemented and 23.23% of power reduction in average has been obtained with 5.28% of propagation delay. Compared with the single-stage Pseudo PG with the transitive fan-in based algorithm, 4.18% more power reduction has been achieved.

Chapter 6 [Conclusions and Future Works] concludes this dissertation on the low power design method called Pseudo power gating and its clustering algorithms. Future works related to the application of Pseudo PG for large scale power aware design are also shown.

Table of contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview of Power Optimization

1.1.1 Basic of Power-efficient LSI Design

With the continuous decrease in feature size of CMOS process and the corresponding increase in chip density and clock frequency of VLSI circuits, we can integrate more than 100 million gates running at several Giga-hertz in a chip, at the same time, we suffer from the significant power consumption. High power consumption is undesirable not only from environmental reasons but also from the heat problem. Excessive heat requires expensive heat removal systems and might destroy the chip in the worst case [1]-[4].

Power and energy efficiency is very important for high end processors and for battery-operated devices, and low power design methodology have been studied hard at every level of abstraction, starting from the semiconductor device level, through circuit and chip levels, all the way to compliers, operating systems. Recently, power is elevated to first class design concern [1]-[4].

To this challenge, in the last several decade, intensive of researches and

works have been done to optimize power consumption at various levels [5]-[6]. The power optimization methods at the different levels of abstraction are briefly introduced in the following parts.

Power is proportional to the square of voltage, clock frequency and the capacitance. So by reducing these parameters, the power can be reduced. Figure 1-1 summarizes power reduction methods clustered at 4 abstraction levels.

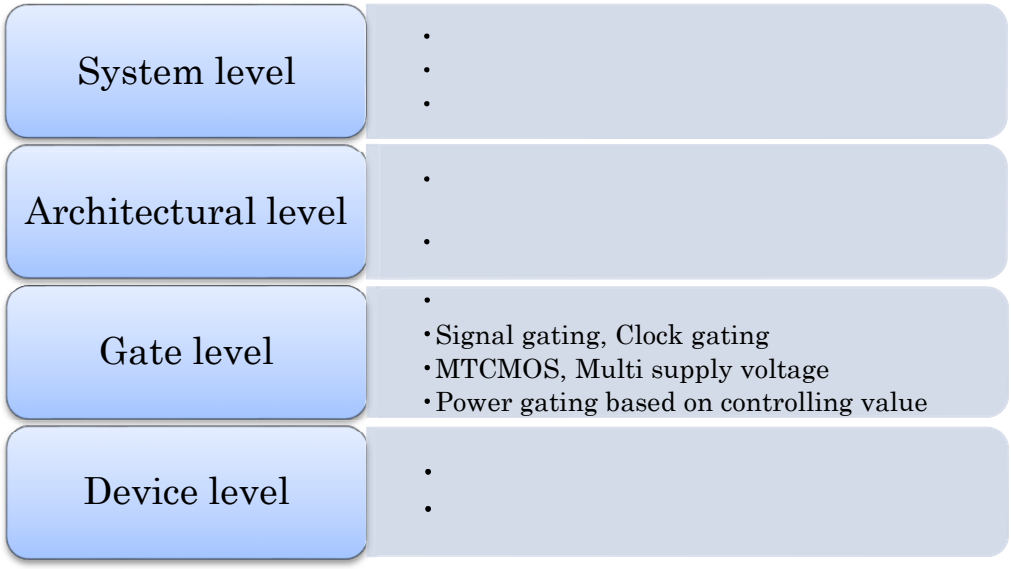


Figure 1-1: Power optimization at different level of design

At the system level, data representation and algorithms selection have been considered for power-efficient LSI design. It is important to making tradeoffs between energy consumption and performance of a design. Main issue is that to develop algorithms which can make the system run at lower frequency and still meet the performance requirements [6]-[8].

At the architectural level, several concurrency increasing transformations are introduced such as loop unrolling, pipelining and control flow optimization as well as critical path reducing transformations [9]-[11] used to allow a reduction in supply voltage without degrading system throughput. A

gray code addressing scheme can be used to reduce the number of bit changes on the address bus.

In gate level, which is also called logic level of abstraction, combinational and sequential functions are implemented using logic elements such as AND, MUX, flip-flops, etc. Logic synthesis converts the register transfer level description to the netlist of gates description. In these two levels, many low power technique can be applicable to general digital systems, including local restructuring, signal gating, clock gating [12]-[20], MTCMOS(Multi-Threshold CMOS), as well as Multi-supply voltage method [21]-[26]. MTCMOS and Multi-supply voltage close to device level. MTCMOS used gate types with different threshold voltage of transistors, and multi-supply voltages uses gates running at different supply voltage with interfacing level converters.

Device level refers to the connection of transistors made from appropriate semiconductor materials and processes of "low-power" cell libraries [27].

While high level (i.e., system level, architectural level) optimizations require dedicated research on specific applications, low level (i.e., gate level, device level) optimizations can be generally applied to many application. This dissertation mainly focuses on the lower levels of abstraction especially on the gate level. In the following the basics of power consumption and dissipation of gates with the operation of CMOS transistors is shown, and power optimization strategies at gate level are summarized briefly.

To achieve low power design at the gate level, consideration about power of a gate constructed from CMOS transistors is necessary and the power analysis of CMOS gates will be shown first.

The power consumption of digital CMOS circuits is generally considered in terms of three components: Dynamic power, leakage power and

short circuit power as shown in Figure 1-2 [2]-[4].

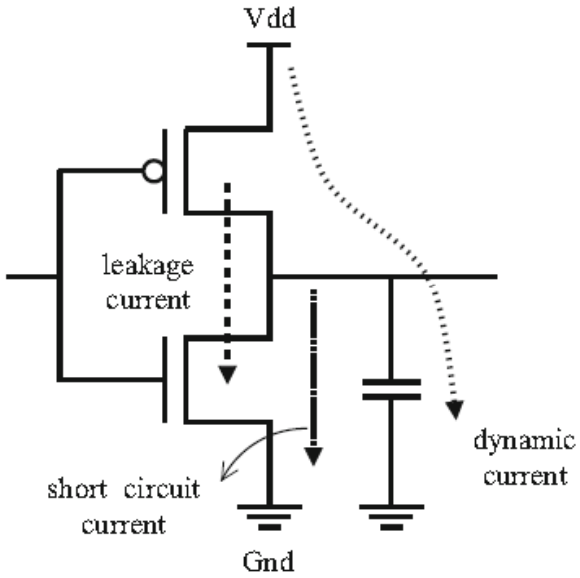


Figure 1-2: Power components of transistor power consumption [2]

Figure 1-2 shows a CMOS inverter (NOT gate) and three power components of the element: $\boldsymbol{P_{dynamic}}$, $\boldsymbol{P_{leakage}}$ and $\boldsymbol{P_{short-circuit}}$.

$\boldsymbol{P_{dynamic}}$: Dynamic power is consumed by charging and discharging of load capacitor when an input changes. Dynamic power is depends on the square of the supply voltage. Details will be shown in Chapter 1.1.4.

$\boldsymbol{P_{leakage}}$: Leakage power is consumed even when the output of the elements does not change. Main component of leakage power is sub-threshold conduction and is caused by current between the source and drain when the transistor is in the sub-threshold region, in which gate-to-source voltage is small than the threshold voltage. Leakage increased total power consumption and it can cause circuit failure when significantly large.

$\boldsymbol{P_{short-circuit}}$: The short-circuit power component is occurred during the transition of the output line (of a CMOS gate) from one voltage level to the

other, there is a period of time when both the PMOS and the NMOS transistors are on, thus creating a path from V_{DD} to ground. This power component is usually not significant in logic design.

In the following subsections, the existing power optimization techniques are discussed on each of different components of power.

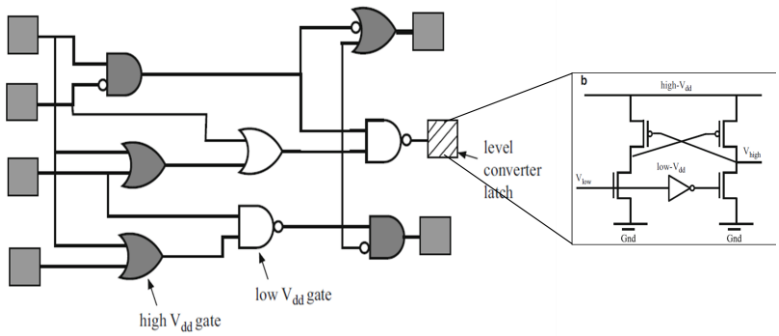
1.1.2 Techniques for Reducing Leakage Power

CMOS gates is widely used because of the complement property of transistor switches, and no current flows in a stable states ideally. However, recent process introduce a leakage current when a transistor is off. Even though a transistor is in a stable logic state, it continues to leak small amounts of power at almost all junctions due to various effects. The subthreshold current for long channel devices increases linearly with the ratio of the channel width over channel length and decreases nearly exponentially with decreasing $V_{GT} = V_{GS} - Vt$, where V_{GS} is the gate bias and Vt is the threshold voltage. Several hundred millivolt of "off bias" typically reduce the subthreshold current to negligible values.

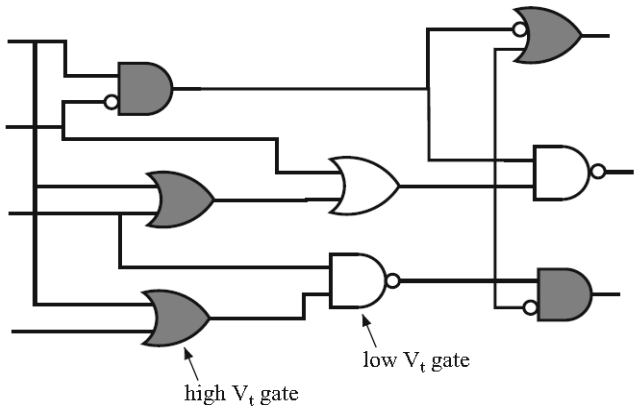
The sub-threshold currents, however, become not negligible with reduced power supply and device threshold voltages. In addition, the sub-threshold current also becomes exponentially dependent on drain voltage at short channel lengths.

Leakage power reduction techniques are especially important for mobile devices such as cellphones, which are "on" but in active mode at almost all of the time. Consequently, leakage power becomes a significant contributor in their power dissipation even if the dynamic power is large. Some of the techniques to reduce leakage power are discussed in the following.

Multiple supply voltages and **multiple threshold voltages method** are effective approaches to reduce leakage power [21]-[26]. The multiple supply system provides a high-V_{DD} for high-performance parts and a low-V_{DD} for low-performance parts. Since the low-V_{DD} might turn off the driven pMOS with high-V_{DD}, a level converter is required between modules with different levels of V_{DD} as shown in Figure 1-3(a). 3 or more supply voltages can be given but the layout becomes complex,



(a) Multiple supply-voltage pipeline stage and level converter latch [2]



(b) Multiple threshold voltage [3]

Figure 1-3: Multiple VDD and multiple Vt method

In the multiple threshold voltages method there provide two or more types of transistors with different threshold. Power can be saved by using a mixture of devices with two or more different threshold voltages. In the simplest

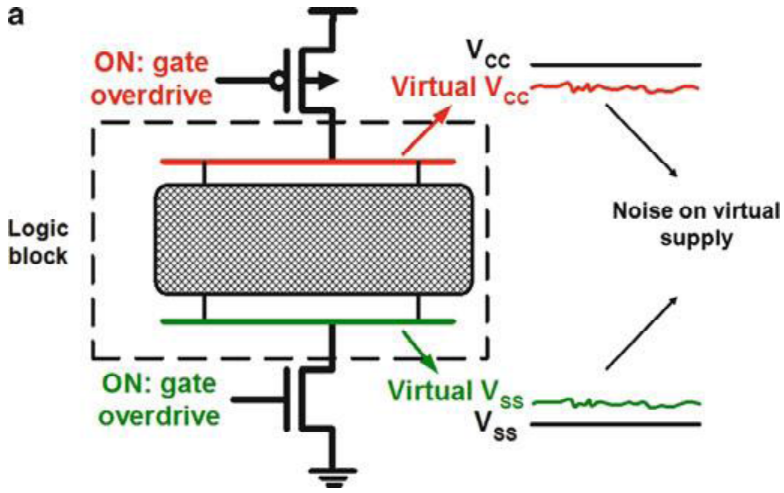
case, there are High-Vt and Low-Vt are available, where Vt stands for threshold voltage. High threshold transistors are slower have less leak, so High-Vt can be used in non-critical circuits. Low threshold transistor have high switching speed but high leak current. They should be used only on critical paths from the point of views of power consumption. Figure 1-3(b) shows an implementation example. Multi-Vt optimization does not change the placement of the cells. However, some additional fabrication steps are needed to support multiple Vt cells, which eventually lengthens the design time, increases fabrication complexity, and may reduce yield.

Power gating method can effectively reduce the leakage power by cutting the supply voltage to the idle logic block or idle logic element. Disconnecting the power supply from the logic can be realized by inserting a sleep transistor as Header or Footer or both. High Vt transistors are usually used as sleep transistors to reduce static leakage power. Low Vt transistor are used in the logic where fast switching speed is important. Logic is supplied by a virtual power rail and sleep transistor are used to connect virtual power rail and supply power rail [28]-[37].

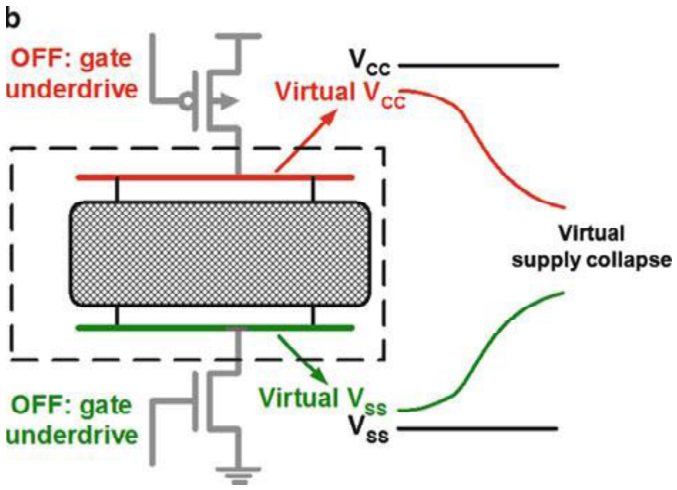
Figure 1-4 illustrates basic structure of power gating which have two modes: active mode (a in Figure1-4) and sleep mode (b in Figuire1-4). Active mode holds in the “on” state and the circuit sees a virtual Vcc and virtual Vss, which are very close to the actual Vcc, and Vss respectively. Idle mode holds in the “off” state, both of the virtual Vcc and virtual Vss go to a floating state. In power gating, however, there are long standing issues over the required external control unit and isolation unit, increase of the rush current especially with larger size of sleep transistor.

[14] and [15] proposed a novel fine-grained CV-based power gating method to avoid these problem. Proposed method does not require the external

control unit and isolation unit, also avoid large rush current by utilizing smaller size of sleep transistor than general power switch. CV-based power gating can realize run-time leakage power optimization because fine grained power control unit (one logic element). The detail of this method will be shown in Chapter 2.1 as related work of this dissertation.



(a) Active mode



(b) Idle mode

Figure 1-4: Power gating structure [2]

1.1.3 Techniques for Reducing Dynamic Power

The short_circuit and subthreshold currents in CMOS circuits are static power consumption without the behavior of the circuit. The dominant source of power dissipation at run-time is the charging and discharging of the node capacitances related to the changes of inputs and output. That is referred as the dynamic power dissipation. One charging event drawn $C_{Load}V_{DD}{}^2$ of the energy proportional from the power source. Among this, about half is charged in the load capacitance and another half is dissipated as heat in pMOS during the charging process.

In a synchronous circuit, inputs of gates changes once per clock, so the dissipated dynamic power is proportional to $C_{Load}V_{DD}{}^2 f$, where f is clock frequency of the circuit. However, the output change of gates depends on the input pattern to the gates. Hence a constant called the activity factor (0<= A <=1) is used to model the average switching activity in the circuit or each gate. The dynamic power of a circuit composed of CMOS transistors can be estimated as:

$$\mathrm{P} \propto \mathrm{A} C_{Load} V_{DD}{}^2 f \qquad (2)$$

Figure 1-5 summarize fundamental techniques to reduce dynamic power based on the equation. Some of these factors should be reduced.

The first fundamental technique is to reduce load capacitance of transistors, $\mathrm{C_{Load}}$. This can be done by using small transistors with low input capacitances in non-critical parts of the circuit. Reducing the frequency of operation f will cause a linear reduction in dynamic power, and reducing the supply voltage $\mathrm{V_{DD}}$ will cause a quadratic reduction.

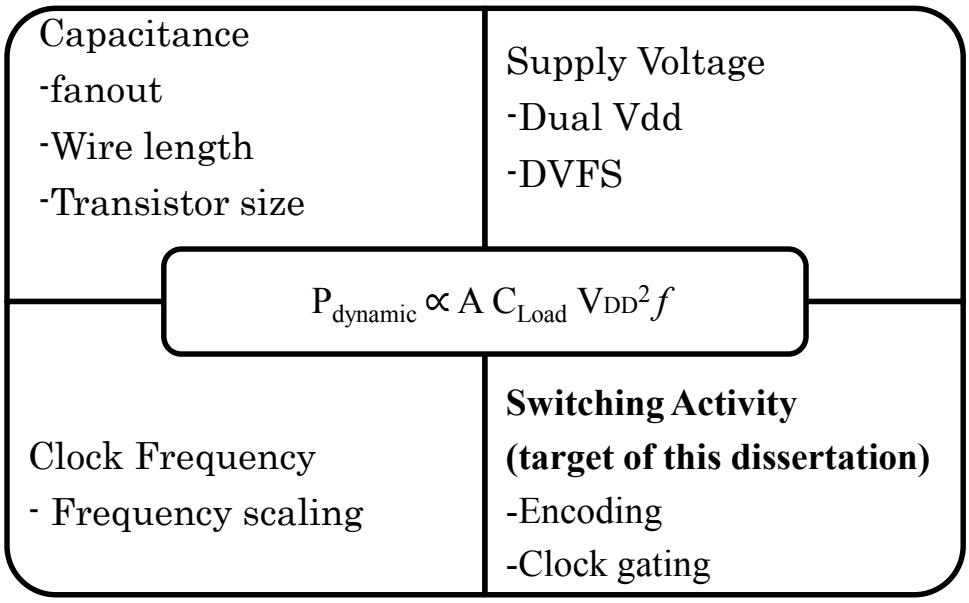


Figure 1-5: Fundamental techniques to reduce dynamic power

However, the delay of a circuit also depends on the supply voltage as follows [20]

$$\tau = k \cdot C_{Load} \cdot \frac{V_{DD}}{(V_{DD}-V_t)^2} \tag{1.3}$$

where τ is the circuit delay, k is the gain factor, C_{Load} is the load capacitance, V_{DD} is supply voltage, V_t is the threshold voltage. The supply voltage reduction achieves quadratic power reduction, but the execution time increases as shown in formula 1.3. The main challenge in achieving power reduction through voltage and frequency scaling is therefore to obtain power reduction while meeting all the timing constraints.

After C_{Load}, V_{DD} and f are decided, the switching activity A remains and that can be measured at gate level. A question come up with this equation is: "how to achieve the same functionality with reduced switching activity?"

To reduce switching activity, clock gating, operand isolation and data-driven signal gating method have been proposed [12]-[20]. In clock gating,

clock to registers is stopped and switching activities of logic gates connected to those registers are reduced. In operand isolation and data-driven signal gating, inputs of some operational module are gated (kept to the same value) by using latches. In such gating methods, extra logic and latches are necessary and the granularity is rather larger. More fine grained method with small overhead is desired.

1.2 Motivation and Contribution of the Dissertation

In this dissertation, we mainly concern about the dynamic power optimization of gate level circuits. Especially we are focusing on methods to minimize the switching activities in combinational circuits.

In a gate level circuit, switching activity is a principal parameter affecting the dynamic power dissipation. In fact, short pulse (called glitches) of gate outputs consume a power a lot. To prevent such glitches, we can use controlling value. Once one input of a logic gate takes controlling value, the value of the other inputs become unnecessary and the computation of the part to the other input waste power consumption. In entire circuit, such power dissipation accounts for a significant amount of overall power consumption. Thus, an effective gate level power optimization strategy which can reduce redundant switching activities is sincerely expected for improving the efficiency of power consumption.

In this dissertation, a Pseudo Power Gating (Pseudo PG) method based on the controlling value is proposed to save such wastage of power computing unnecessary input values. Distinguished from other power gating methods, the proposed power saving mechanism is capable of reducing dynamic power consumption efficiently by optimizing the switching activities in ultra-fine grained region without extra hardware circuitry. We will show the gate level

optimization method to reduce the switching activity using the proposed Pseudo PG method.

To implement Pseudo PG, the clustering algorithm that can automatically identify optimum control signal and corresponding power-controlled blocks is essential. In this dissertation, we analyze the relation between switching activity of controlled gates and controlling probability to explore how to take full advantage of feature of CV. Consequently, a heuristic algorithm switching activity first (*SW*-first) algorithm is proposed to maximize the reduction of switching activity of the whole circuit. In order to get the maximum power reduction, we relate the power optimization to the glitch problem and propose a simple method to prevent the occurrence of glitches.

Considering the possible depth increasing at the insertion of control signal, we are dedicated to maintain the performance of circuit by using steady maximum depth constraint. By introducing proposed pseudo power gating with *SW*-first algorithm, on averagely 13% of power consumption can be reduced in benchmark circuit under the maximum depth constraint. Further experiments proved that the proposed method is also effective under the different statistic of primary input.

Although SW-first algorithm makes Pseudo PG effectively running at the combinational circuit, the steady maximum depth constraint, unfortunately, limits the potential power reduction within the abandoned part.

To overcome this limitation and achieve further power saving, we propose new clustering method as first improvement. Newly proposed method based on an extension of controlling value of a gate, to improve the clustering mechanism for one decided control signal. The controlling value of an input of a gate can disable the power of blocks computing other inputs. In the new method, not only the direct input but also the transitive fan-ins are considered if the

fan-in's controlling value becomes the controlling value of the focusing gate. Transitive fan-ins have the smaller depth and can be the control signal to the blocks of other inputs.

The experimental results show that 19.05% average power reduction has been achieved by applying proposed new clustering method. Approximately 6% of power is reduced from the circuit only applied SW-first algorithm.

Although the above improvement has achieved good performance on power reduction, it is still hard to stop the unnecessary computation of power-controlled block completely. Especially for the largely clustered block, there exists unnecessary computation when the current control signal takes non-controlling value. This situation limits the further reduction of power consumption. Thus, as the second improvement, a new dual-stage structure is proposed to realize further switching activity reduction inside the power-controlled blocks, when control signal takes non-controlling value. Dual-stage structure is realized by adopting another Pseudo PG to the inside of power-controlled block to stop unnecessary computation

Then, the proposed dual-stage Pseudo PG structure is further implemented and 23.23% of average power reduction has been obtained with acceptable propagation delay.

1.3 Organization of the Dissertation

The rest of this dissertation is organized as follows.

Chapter 2 [Preliminaries] explains fundamental concepts such as controlling value based power gating and Binary Decision Diagrams (BDDs). The switching activity estimation using BDD is also given in this Chapter.

Chapter 3 [Pseudo Power Gating Method Based on Switching Activity

First Algorithm] presents the basic idea and architecture of Pseudo Power Gating method. To automatically implement this method to logic circuit, two clustering algorithm are introduced in this chapter. Occurrence of glitches and depth increase by inserting control signal are resolved effectively by executing additional constraints. Specific mechanism of these constraints will be introduced in this chapter.

In Chapter 4 [Transitive Fan-in based Clustering Algorithm for Further Power Saving] we delve in to the detailed power saving mechanism and point out some potential power reduction opportunities limited by depth constraint. We also propose new clustering method based on the extension of controlling value to achieve further power reduction.

In Chapter 5 [Dual-stage Pseudo Power Gating] we show that potential power reduction opportunities in the power-controlled blocks. We propose Dual-stage structure to achieve further power reduction in the already clustered blocks. The simple realization method of Dual-stage structure is discussed.

In Chapter 6 [Conclusions and Future Works], we conclude this dissertation by summarizing the proposed Pseudo Power Gating and corresponding clustering algorithms as well as the approach of Dual-stage structure. Furthermore, some future works such as application of proposed method in large score of circuit are addressed.

Chapter 2

Preliminaries

2.1 Power Gating Based on the Controlling Value

Controlling value (CV) of a logic gate is a logic value which can decide the output of this gate by one input taking the value regardless what values the other inputs take. Hence, if one of the inputs takes controlling value, the computations of the blocks which generate other inputs are unnecessary. To save the power consumption of such unnecessary computations, as introduced Chapter 1.1, the novel fine grained power gating method based on controlling value (CV-based power gating) is proposed in [14]-[15] to achieve run-time leakage power reduction.

The basic idea of CV-based power gating is to use one input of a logic gate to power-off the logic blocks generating other inputs without external control circuitry and signal isolation cells. Figure 2-1 shows the basic structure of CV-based power gating by illustrating an AND gate case. In this figure, an *nMOS* type transistor is used as the sleep transistor and inserted between the ground line and the logic block. The sleep transistor is simply controlled by signal *b* which is an input signal of the current AND gate. If signal *b* takes the

controlling value logic-0, the sleep transistor is turned off and the logic block 1 will be cut off from the ground line, by this, *a* might be unknown value. While the final output is determined to logic-0 only by signal *b*. Otherwise, if signal *b* takes logic-1, the sleep transistor is turned on simultaneously and the logic block 1 is work normally. The output signal is decided by both input signals *a* and *b*. For NAND gate, since the controlling value is also logic-0, the architecture of CV based power gating is the same as shown in Figure 2-1. For the case OR gate or NOR gate, a pMOS transistor can be inserted to supply line and logic we want to control to cut off the power by connecting control signal with controlling value logic-1.

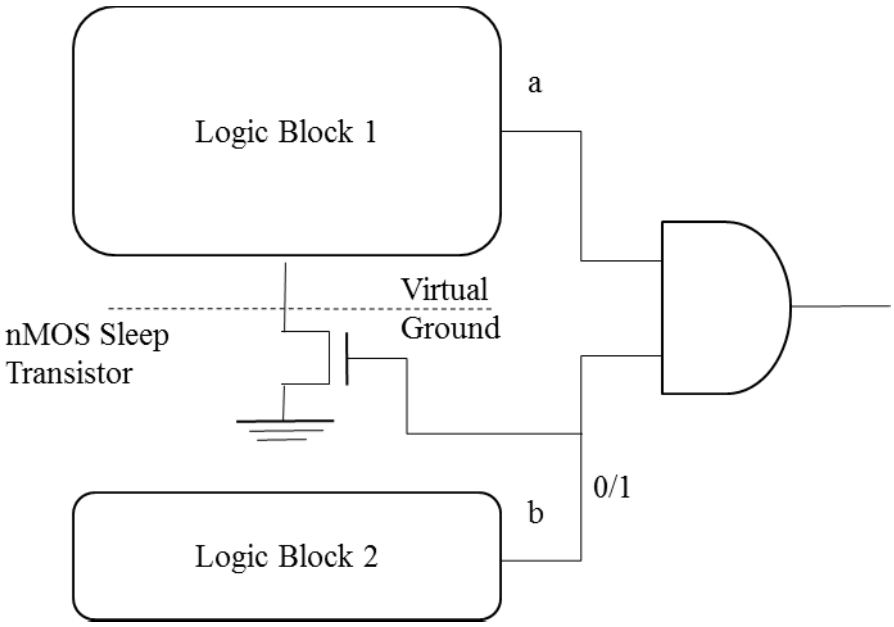


Figure 2-1: CV-based power gating structure [14]

The most notable advance of this method is that the external control unit and isolation unit are not required so that the design area and additional energy to drive those external devices are not more needed.

The effectiveness of the CV-based power gating depends on the probability of taking the controlling value and also the number of gates controlled

by the sleep signals. The product of the probability of taking the controlling value and the number of gates is called the “expected number of sleep gates" and is used to evaluating the results of the CV-based power gating method. Several heuristic algorithms are proposed in order to achieve considerable power reduction by properly selecting sleep signals and power clusters.

Simulation results shows that, by applying the CV-based power gating method, up to 27% of total power reduction on the ISCAS’85 standard benchmarks is achieved. The [15] pointed that, obtained power saving mainly consist of dynamic power since the simulation utilized 180nm process in which leakage current is nearly negligible. Although the simulation results are very considerable in the gate level power optimization, however, implementing hundreds of sleep transistor for ultra-fine grained power gating is difficult to realize in practice. An undeniable fact is that, by utilizing property of the controlling value to control the unnecessary computation is very useful method for power saving. In this dissertation, controlling value is also the foundation of the power saving mechanism in the proposed Pseudo power gating method.

2.2 Switching Probability Estimation

Since that switching activity of combinational circuit is the main target of this research, accurate estimation of switching activity of CMOS circuits is important.

Methods of estimating the activity factor A at a circuit node n involve estimation of signal probability $prob(n)$, which is the probability that the signal value at the node takes logic value 1 under the assumption that the values applied to each circuit input are temporally independent. The activity factor A can be written as:

$$A = 2\, prob(n)(1\text{-}prob(n)) \tag{2.1}$$

Note that A corresponds to the probability changing turn 1 to 0 or turn 0 to 1. $prob(n)$ is 1-probability, (1-$prob(n)$) is 0-probability. Computing signal probabilities has attracted much attention. As earliest work, signal probabilities in a combinational network are computed by algebraic expressions. If a network consists of simple gates and has no reconvergent fanout, then the exact signal probabilities can be computed during a single post-order traversal of the network using following equations:

$$\text{NOT gate:} \quad prob(o) = 1 - prob(i),$$

$$\text{AND gate:} \quad prob(o) = \prod_{i \in inputs} prob(i),$$

$$\text{OR gate:} \quad prob(o) = 1 - \prod_{i \in inputs} (1 - prob(i)).$$

However, for networks with reconvergent fanout, it is difficult to compute exact signal probabilities.

In this dissertation, the Binary Decision Diagrams (BDDs) is used to parse the netlist of target circuits and to describe the structure of the circuit for further variation of circuit. Furthermore, BDD can also be used to calculate the logic-0 and logic-1 probabilities with high accuracy. Following part introduced general explanation about BDD and the method to calculate signal probability using BDD.

2.2.1 Binary Decision Diagrams (BDDs)

A Boolean function $f : B^n \rightarrow B$, B={0,1} can be represented by the following formula commonly known as the Shannon Decomposition[38]-[39].

$$f = \overline{x_i} f_0 + x_i f_1 \tag{2.2}$$

Where f be a Boolean function defined over the variable set $X = \{x_1, \dots, x_n\}$ and f_0 denotes the cofactor of f with respect to $\overline{x_i}$ and f_1

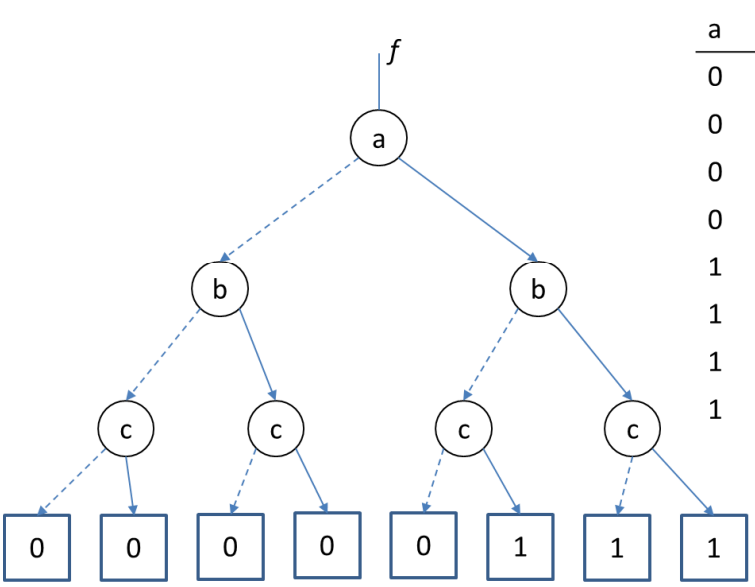
denotes the cofactor of f with respect to x_i. A BDD can be used to represents a Boolean functions as a rooted, Directed Acyclic Graph (DAG) with a vertex set containing two types of vertices, non-terminal and terminal vertices. A non-terminal vertex is also called internal node and has two attributes: (i) An argument index $v \in \{x_1, \dots, x_n\}$. (ii) Each internal node has two existing edges that point to cofactor subgraphs f_0 and f_1 respectively. f_0 is commonly called as low child and f_1 is called as high child. There are two types of terminal vertex, 0-terminal and 1-ternimnal and they has no outgoing edge.

Figure 2-2 shows a binary decision tree and a truth table representing the function f (*a*, *b*, *c*). In this tree, the value of the function can be determined for a given variable assignment by following a path from the root to a terminal. In the figure, dotted lines represent edges to a low child, while solid lines represent edges to a high child. Therefore, to find (*a*=1, *b*=1, *c*=1), begin at *a*, traverse down 3 solid lines. This leads to the terminal 1, which is the value of f (1, 1, 1).

It can be observed that, if both child of a node lead to same vertex then the node can be removed from the tree to save the node resources. For instance, all branches from the low child of *a*, f(0,0,0), f(0,0,1), f(0,1,0) and f(0,1,1) in Figure 2-2, are leads to 0-terminal, so *a* can directly leads to 0-terminal with dotted line. By repeating to merge isomorphic subgraphs and eliminate the node whose two child are isomorphic, we can get reduced BDD as shown in Figure 2-3. In a reduced BDD, there exist no two isomorphic subgraphs, which represent the same Boolean function. In many case, reduced BDD is expected for the readability and high utility of resources.

In the other hand, the number of BDD node is determined by chosen ordering of the variables. For a Boolean function, we might end up getting a graph whose number of nodes would be linear at the best and exponential at the worst case. In an ordered BDD, each variable can occur only once on each path

and in the same order for any possible path. The term BDD usually refers to the indicated the reduced ordered BDD (ROBDD) in the literature.



a	b	c	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Figure 2-2: Binary decision tree and truth table for Boolean function f=a(b+c)

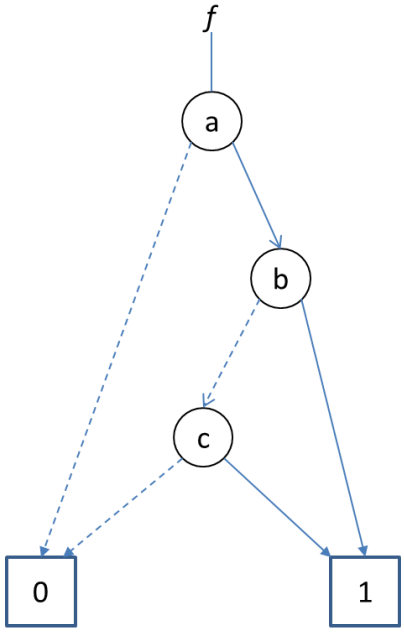
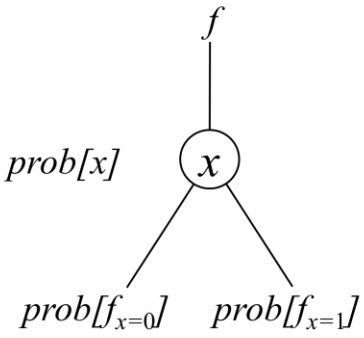


Figure 2-3: Reduced BDD of Boolean function f

In this dissertation, the BDD is used to describe the structure of the benchmark circuit for further calculation. A single graph can be used to represent both *f* and $\overline{f}$, where the latter functions is identified by a complement attribute on the incoming edge. Furthermore, BDD can also be used to calculate the logic-0 and logic-1 probabilities with high accuracy.

2.2.2 Probability Computation from BDD

An output probability of a function, *f*, denoted as *prob*[*f*], is the probability that f takes logic-1 at some arbitrary time of observation. Consider a function *f* having the probability *prob*[*x*] for the input variable *x* and the output probabilities *prob*[$f_{x=0}$] and *prob*[$f_{x=1}$] for the corresponding cofactors $f_{x=0}$ and $f_{x=1}$. In terms of a BDD, this relationship is shown in left in Figure 2-4.



$f_{x=0}, f_{x=1}$ \ x	0	1
0 0	0	0
0 1	0	1
1 1	1	1
1 0	1	0

Figure 2-4: 1-probability of BDD vertex.

From truth table of *f* shown in right of Figure2-3, *f* take logic value 1 when the variable x and cofactor $f_{x=1}$ are 1-valued or when the variable *x* is 0-valued but the $f_{x=0}$ is 1-valued. $f_{x=0}$ and $f_{x=1}$ are mutually exclusive, so the 1-probability of the formula *f* is:

$$prob[f = 1] = prob[x] \cdot prob[f_{x=1}] + (1 - prob[x]) \cdot prob[f_{x=0}] \quad (2.3)$$

If the variable *x* has same property 0.5 to take logic-1 and logic-0 then 1-probability of the formula *f* became:

$$prob[f = 1] = 0.5 \cdot prob[f_{x=1}] + 0.5 \cdot prob[f_{x=0}]$$

$$= 0.5 \cdot (prob[f_{x=1}] + prob[f_{x=0}]) \quad (2.4)$$

In this formula, $f_{x=1}$ and $f_{x=0}$ represent high child and low child subgrapgh respectively. A similar statement can be made for the 0-probability of the formula *f*:

$$prob[f = 1] = 0.5 \cdot ((1 - prob[f_{x=1}]) + (1 - prob[f_{x=0}])) \quad (2.5)$$

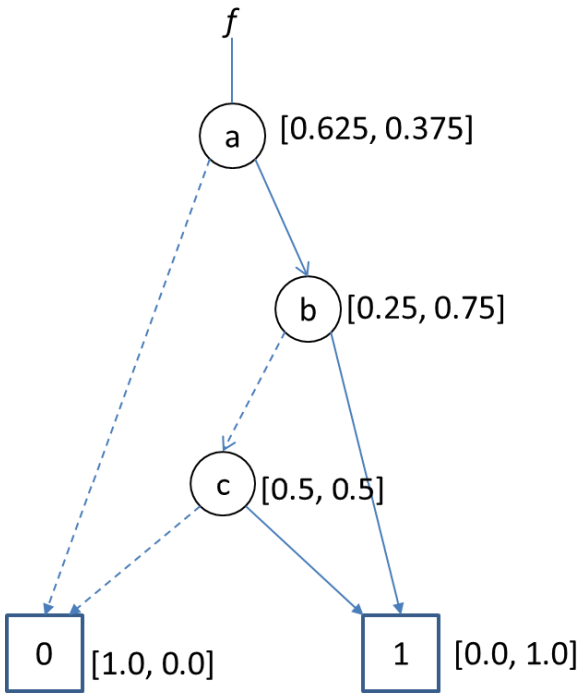


Figure 2-5: Probability computation using BDD

In this dissertation, input patterns utilized to evaluate the proposed power optimization methods are given randomly so that the 1- and 0-probability of input variable can be considered as 0.5 respectively. Therefore, the probability of each signal in a circuit, or in other words the probability of each sub-BDD in a BDD can be obtained by iteratively computing formula (2.5) until reach the 1- or 0-terminal vertex. Since 1-terminal represents a variable

assignment for which the represented Boolean function is true, 1- and 0-probability can be considered as 1.0 and 0.0 respectively. For 0-terminal, opposite value 0.0 and 1.0 represent the 1- and 0-propobility.

We still use the Boolean a function f(a,b,c) = a(b+c) as an example, probability computation results using BDD is shown the Figure 2-5. 0- and 1-probability are shown nearby each vertex such as [0.0, 1.1] for 1-terminal node.

Chapter 3

Pseudo Power Gating Based on SW-first Clustering Algorithm

3.1 Overview

Various power estimation and optimization techniques have been proposed and plenty of library modules and synthesis tools for low power design have been developed. Although supply voltage reduction is desirable with its effectiveness, it suffers from the tradeoff between speed and power. Hence, it is also important to develop power reduction technique at the logic gate or transistor level. In the logic level design, switching activity is the principal parameter affecting the dynamic power consumption of a circuit. A data-driven signal gating method is proposed in [12] to prevent the unnecessary switching activities. However, this method requires an extra hardware such as latch control unit which causes additional area and delay. Operand isolation scheme is another method to reduce redundant switching in datapaths. However, extra hardware units are required such as AND, OR gates or latches which not only

incur considerable overhead in terms of delay, power, and area, but also make the implementation of logic for automatic selection of latches difficult [13]-[17]. Clock gating [18]-[20], is one of the effective method to reduce the dynamic power dissipation by stopping the clock pulse to unnecessary register and circuit blocks. Clock gating has been frequently used in reducing the power consumption of registers and the clock network whose effectiveness depends on control signal for clock gating. Clock gating has the relation to power gating, if a register is clock gated then its input can be power gated. There are several works, on such run-time power gating.

In this chapter, we propose a novel method for optimizing switching activity in ultra-fine grained region with no extra hardware unit. The target of the proposed pseudo power gating (Pseudo PG) is to reduce power consumption of combinational part [40]-[41]. Conversely, clock gating is more reduces mainly the power of registers in sequential circuits. If a sequential circuit has been applied clock gating already, the proposed pseudo power gating can also be applied to combinational part to reduce further power consumption.

The proposed method used controlling value of a logic element. For each 2 input logic element having controlling value, there exists a sub-module controlled by an input to the element. The power reduction depends on the order of the sub-module selection or the control signal selection. A basic algorithm has been developed to implement the proposed method to circuit automatically to check its effectiveness. By analyzing actual switching activity reduction after inserting control signal, SW-first algorithm is also proposed to improve the power saving capability.

In the application, steady maximum depth constraint is introduced to prevent the depth increase caused by the insertion of the control signal. We also checked various factors affecting the power consumption of gate level circuits

when applying the Pseudo PG. One of such factors is the occurrence of glitches by applying Pseudo PG. Glitches increase the power consumption a lot and a method to reduce the occurrence of glitches is proposed by considering the parity of inverters. The proposed Pseudo PG method was implemented in C language and evaluated on ISCAS'85 benchmark, using SPICE simulation for the netlist extracted from the layout result with the VDEC Rohm 0.18μm process.

The average power consumption of ISCAS'85 circuits is reduced to 13% under the acceptable propagation delay overhead. Finally, the effectiveness of proposed method under the different statistic of primary input is considered.

The rest of this chapter is organized as follows: Section 3.2 gives a general introduction of proposed pseudo power gating method. Section 3.3 describes the–clustering algorithm and the methods to prevent the increase of maximum depth and the occurrence of glitches. Section 3.4 shows the experimental results and discusses the case that pseudo power gating under the different statistics of primary inputs. The conclusion is given in Section 3.5.

3.2 Pseudo Power Gating Method

Chapter 2.1 introduced CV-based power gating method which realizes ultra-fine grained power gating based on the controlling values of logic elements. In this method, unnecessary computation part is efficiently power-controlled by inserting the sleep transistor controlled with a signal which takes the controlling value on a logic gate. A power controlled block is connected to a sleep transistor in CV-based power gating, and several logic gates might share a virtual ground controlled by the sleep transistor. Its effectiveness depends on the probability under which the control signal takes the controlling value.

In other words, if one input of a logic gate takes controlling value, the switching activity inside the part generating other input is unnecessary and the power gating mechanism can be considered to save those toggling effects.

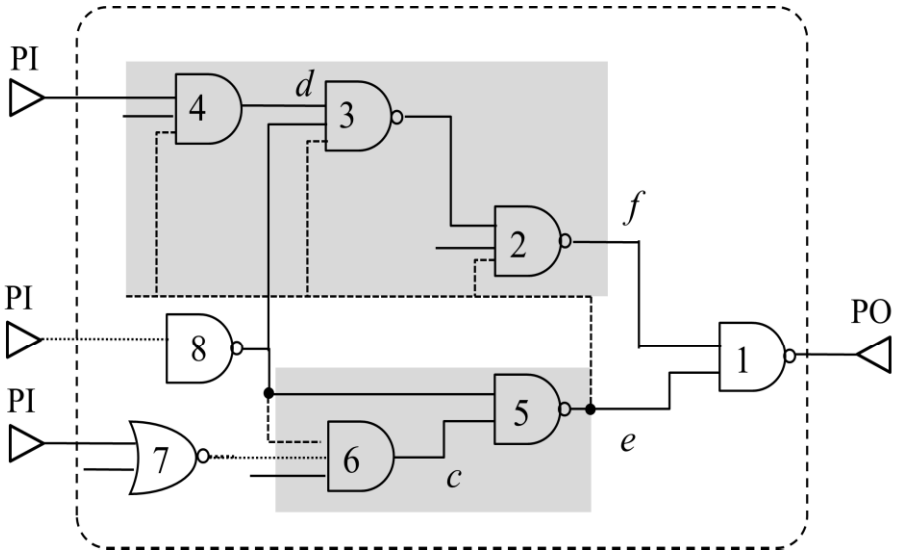
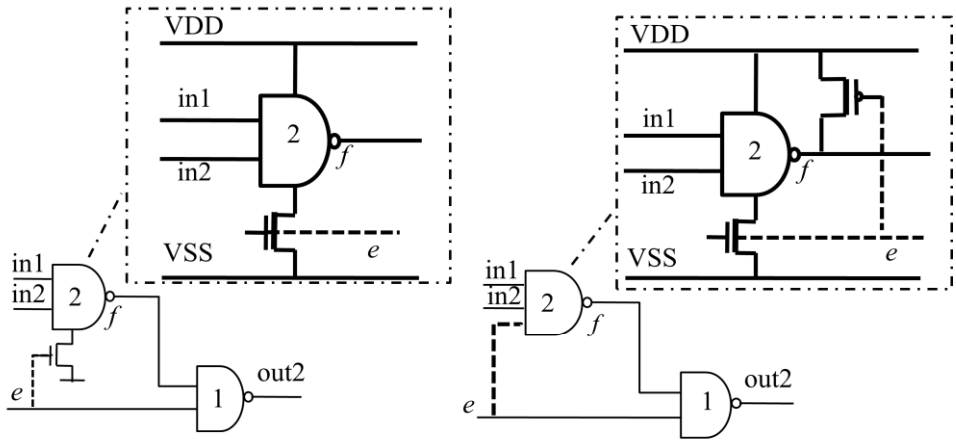


Figure 3-1: Pseudo power gating structure.

To enhance the CV-based power gating to an ultra-fine grained signal gating region at the gate level, the pseudo power gating method is proposed to reduce the dynamic power consumption of a circuit by controlling unnecessary switching activity. The gate level optimization method will be shown to reduce the switching activity using the proposed Pseudo PG method.

The basic idea of the Pseudo PG method is to stop the switching behavior of an unnecessary computation part relating to a logic element having the controlling value on one input. The unnecessary part corresponds to the computation for side input relating to the input taking the controlling value. It is effective to stop the output change of each gate in such unnecessary computation module for reducing the dynamic power. To achieve this objective, the control signal is manipulated as an extra input for the gates to be controlled

just like signal *e* in the example shown in Figure 3-1. For gates 2, 3 and 4 in the shaded area, signal *e* is inserted as additional input to control the output changes of these gates. Figure 3-2 shows the difference between transistor level structure of CV-based power gating [14] and proposed Pseudo PG, where one more pMOS transistor is required comparing to CV-based power gating. The Pseudo PG can more easily be implemented under usual synthesis flow, and the Pseudo PG can be escaped from the rush current problem.



a) CV-based power gating b) Pseudo power gating

Figure 3-2: Difference of transistor level structure of two methods

If we focus on the gate 1 of Figure 3-1, the input signal *f* also has opportunity to control gate 5 and 6. However, logical collision may occur if both *e* and *f* act as control signal to control the block generating each other. This gives rise to an important issue with the Pseudo PG approach: How to select the control signal to obtain maximum power reduction? To deal with this issue, optimum clustering algorithm is proposed to meets switching activity optimization demand.

In addition, extra consideration is required for a gate with fan-outs and a gate has different CV with control signal. The fan-outs from the

controlled-gate, except the one which flow into the reconvergent gate, will make difference to primary output due to the changed functionality of the controlled gate. Gate 8 in Figure 3-1, for an instance, has two fan-outs flow into gate 3 and 5 respectively. Gate 8 cannot be controlled by any control signal directly, but can be controlled by using an OR gate which gathers the signal *c* and *d* simultaneously. Obviously, it can dramatically increase the complexity of circuit. Thus, controlling a gate with fan-outs is not suitable for clustering algorithm. Similarly, A gate has opposite CV with control signal cannot be directly controlled by control signal. For example, Gate 7 cannot be controlled by signal *b*, but can be controlled by inverted signal b. Penalty increasing by required extra inverter is the main reason for giving up controlling the gates with fan-outs or gate with different CV.

In next section, the relation between switching activity of controlled gates and controlling probability is analyzed to explore how to take the full advantage of the feature of CV. A heuristic algorithm is proposed to enlarge the reduction of switching activity of the whole circuit. In order to get the maximum power reduction, we relate the power optimization to the glitch problem and propose a simple method to prevent the occurrence of glitches. On the possible depth increase with inserting control signal, a steady maximum depth constraint is shown.

3.3 Clustering Algorithm for Selecting Optimal Control Signal

3.3.1 Basic Clustering Algorithm of Pseudo Power Gating

Before introducing a clustering algorithm for the proposed Pseudo PG,

it will be explained the previous clustering algorithms for CV-based power gating as a basis. The methods maximize the probability to power off charged gates using several heuristic algorithms [14], [15]. The value of $\sum_{i=1}^{N_{domain}} N_i \times P_i$ (Expected sleep gates N_E) is evaluated as expectation of power savings and algorithms are proposed to cater to maximize the value of N_E.

The basic algorithm is called as *N*-based algorithm, where *N* abbreviates the number of controlled gates. At first, the algorithm sorts power-off blocks with the total number of gates in the blocks, and select the power-off blocks one by one from larger number to smaller number of gates. Assume that the number N_i of gates in one power-off block is power controlled by signal i, and the probability of taking the controlling value (hereafter: controlling probability) is P_i. Basic algorithm maximizes the value of $\sum_{i=1}^{N_{domain}} N_i$, where N_{domain} presents total number of power-off blocks in the circuit. Probability-first heuristic algorithm sorts the blocks with the controlling probability, and selects the power-off block one by one in descending order of the probabilities to maximizes the value of $\sum_{i=1}^{N_{domain}} P_i$.

Another heuristic algorithm is proposed in [15] considering the gate count N_i and the controlling probability P_i simultaneously. That is called *pN*-based algorithm, where *pN* is the abbreviation of the multiplication of the probability and the gate count. At first, the highest $N_i \times P_i$ block is selected. Then the second highest one is selected, and so on. Experimental results show that, the *pN*-based algorithm outperforms other algorithms for the CV-based power gating method.

For the proposed Pseudo PG method, as the basic algorithm, *N*-based is applied to evaluate the efficiency of the method in some benchmark circuits. Typically, the gate type constraints should be added to *N*-based algorithm to maintain the accurate functionality of the circuit.

By applying the *N*-based algorithm, a control signal is decided for each controlled gate, and the control signal is manipulated as an additional input in the implementation. Note that, the value of switching activity of controlled gates is varied due to the changing probability of the outputs. Subsequently, the switching activity of all gates on the transitive fan-out cone will be varied and it could not ensure that the reduction of switching activities of whole circuit. Therefore, the clustering algorithm with considering actual variation of switching activities is necessary. By taking into account the chain reaction of switching activities, feasible clustering algorithm for the Pseudo PG method is presented in next section to maximize the power savings of the circuits.

3.3.2 Switching Activity Analysis with Controlling Probability

In this section, the relation between switching activity of a controlled gate and controlling probability are described. Switching activity at a gate is defined as the sum of the probability of the output transition from 0 to 1 and that from 1 to 0 of the gate. The transition probability of a gate can be computed as the product of the probability that gate taking 0 and taking 1. Switching activity of a signal can be evaluated by $2 \cdot P \cdot (1 - P)$, when P and $1 - P$ represents the probability of a logic-1 and logic-0 occurring at the output of the gate respectively.

Figure 3-3: Probability and switching activity variation

Using a NAND gate in Figure 3-3, the relation between controlling probability and switching activity of its output before and after adding an extra input is shown. Let P_a, P_b and P_c denote the probability of non-controlling value (1-Probability) of inputs *a*, *b* and control signal *ctrl*. Thus 1-P_c presents controlling probability of control signal. Assume that the two inputs and control signal are independent of each other, switching activity of the output before the controlling can be represented by:

$$SW_{before} = 2 \cdot P_{before} \cdot (1 - P_{before}) = 2(1 - P_a P_b)(P_a P_b) \quad (3\text{-}1)$$

After the controlling, switching activity becomes:

$$SW_{after} = 2 \cdot P_{after} \cdot (1 - P_{after}) = 2(1 - P_a P_b P_c)(P_a P_b P_c) \quad (3\text{-}2)$$

Use α to substitute $P_a P_b$, then the expression becomes:

$$SW_{before} = 2(\alpha - \alpha^2)$$

$$SW_{after} = 2(-\alpha^2 P_c^2 + \alpha P_c) \quad (3\text{-}3)$$

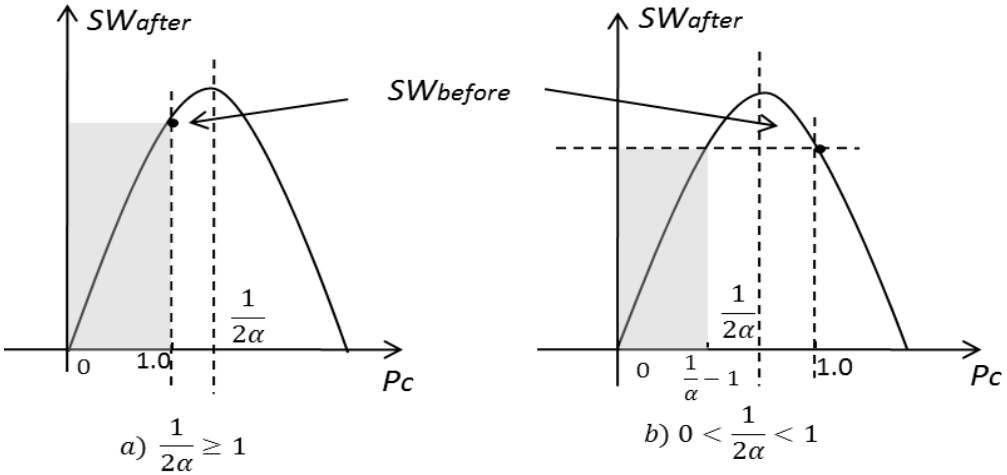


Figure 3-4: Switching activity and controlling probability

The graphical representation of this quadratic equation is shown in Figure 3-4. The parabola of the expression opens downwards and the vertex is maximum value when P_c takes $\frac{1}{2\alpha}$. And, P_c takes 1 means that controlling probability is 0, so corresponding vertical axis value represents the switching activity before the control signal insertion (black spots on the curve).

As shown in Figure 3-4 a), if $\frac{1}{2\alpha} \geq 1$, switching activity of the circuit with control signal is always smaller than that of the original circuit. On the other hand, if $0 < \frac{1}{2\alpha} < 1$ as shown in Figure $3-4$ b), switching activity of the circuit with control signal might be larger than that of the original circuit depending on the value of P_c. If P_c is less than $\frac{1}{\alpha} - 1$, then the switching activity can be reduced. However, if P_c is larger than $\frac{1}{\alpha} - 1$, the switching activity becomes larger by adding the control signal. These cases are shown in the following (3-4).

$$\begin{cases} \text{case 1,} \quad \frac{1}{\alpha} - 1 \leq P_c < 1 : \ Pseudo\ PG\ is \text{ uneffective} \\ \text{case 2,} \quad 0 < P_c < \frac{1}{\alpha} - 1 : \ Pseudo\ PG\ is \text{ effective} \end{cases} \tag{3-4}$$

Therefore, the control signal insertion should be checked based on the switching activity reduction as follows:

$$\begin{aligned} Reducion_{SW} &= SW_{before} - SW_{after} \\ &= 2(\alpha^2 P_c^2 - \alpha P_c + \alpha - \alpha^2) \end{aligned} \tag{3-5}$$

The control signal can be added only when the value is larger than 0. This is used in the following algorithm.

The formula (3-5) shows the reduction of the switching activity of one gate. In reality, the switching activity of the fan-out gates is also affected, and the switching activity of all gates in the transitive fan-out cone of the gate with the

control signal is considered. Instead of focusing on maximizing the parameter $\sum N_i$, $\sum P_i$ or $\sum N_i \times P_i$, maximizing the actual reduction of switching activities of all related gates is more appropriate for the Pseudo PG method. Accordingly, we choose a cluster which maximizes $\sum Reducion_{SWi}$.

3.3.3 Switching Activity First Algorithm

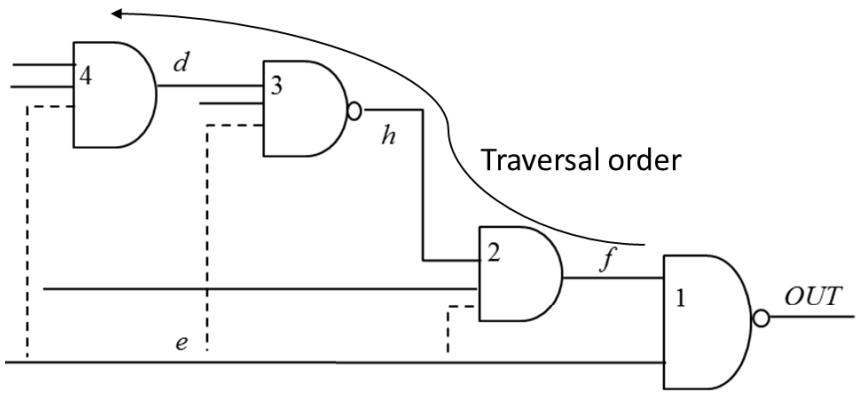
The switching activity first algorithm (*SW*-first algorithm) is proposed with considering the chain reaction of switching activity of controlled gates.

The basic strategy of *SW*-first algorithm is to apply the Pseudo PG from the power domain with the highest reduction of switching activity of whole circuit. Partial key sub-procedure of *SW*-first algorithm is shown in ***Algorithm 1***. To implement *SW*-first algorithm, both logic-1 and logic-0 probabilities for each signal in the original circuit are computed by employing a BDD based method. Probabilities of primary inputs are assumed to be 0.5 and switching activity for each signal are calculated by $2 \cdot P_0 \cdot P_1$.

Next, a traversal program is performed to check the possible pair of control signal and power controlled blocks. During backward traversal, both of two inputs of each gate in the circuit have opportunity to become control signal to control the block generating the other input. Thus, once the program to reach a gate in the circuit, switching activity reduction related to each of two inputs is calculated and the input leading to better reduction is selected as a control signal.

More particularly, when a gate i is traversed by the program, for one of its inputs (IN1(i)), its controllable gate can be searched in the block generating input clustered (IN2(i)) and be clustered as one block named power-controlled block of control signal (IN1(i)). Whenever one controllable gate was found during the backward tracing IN2(i), probability of this gate should be recomputed based on the probability of control signal in here.

The probability of the gates in the transitive fan-out cone should also be recomputed since that some of these gates might be controlled in future traversal. Specific recomputation mechanism is shown as follows: in the clustering algorithm process, for one controllable gate in charged to control signal, the BDD of the output node should be renewed to cover the inserted control signal. After that, the procedure computes the 1-probability of the gates affected by control signal insertion. In fact, the computation is not too much. This is because the logic function of reconvergent gate, which cascades both controlled gate and control signal, is not changed. In other words, the probabilities from reconvergent gate to primary output are not changed.



SW of output / *e inserted to*	*d*	*h*	*f*	*OUT*
Gate 2	-	-	recompute	-
Gate 3	-	recompute	recompute	-
Gate 4	recompute	recompute	recompute	-

Figure 3-5: Switching activity update after control signal insertion

For instance, in Figure 3-5, assume that signal *e* a candidate control signal of Gate 2, 3 and 4. The switching activity reduction should be computed to evaluate the candidate control signal *e*. Based on the depth first traversal, Gate 2 is reached first and switching activity change of the signal *f* are checked by updating probability of *f*. If switching activity reduced by passible inserting of signal *e* is larger than 0, Gate 2 can be kept in power-controlled block of signal *e*. After that, Gate 3 is reached, the switching activities change of signal *h* and *f* are checked and accumulated as one value to represent the switching activity reduction of the control signal insertion to Gate 3. Similarly, for Gate 2, switching activity change of signal *d, h, f* are accumulated as one value to evaluate the control signal insertion to Gate 2. Finally, results of three insertion are accumulated to one value, to evaluate the power saving efficiency of signal *e*.

After the evaluation of candidate control signal IN2(*i*) explained so far, the same procedure is executed on IN1(*i*) to recompute the switching activity of each gate in the circuit modified by possible control signal IN2(*i*).

Note that the recomputation of switching activities in both two possible assignments of control signal, IN1(*i*) and IN2(*i*), should be completed with same circuit status formed by latest control signal insertion. Let *SUM_sw*(*j*) denotes the accumulation of switching activity in the latest status of circuit, and *reduction_1*(*i*) represents the switching activity reduction obtained by inserting control signal IN1(*i*), *reduction_2*(*i*) represents switching activity reduction obtained by inserting control signal IN2(*i*).

Algorithm 1 Pseudo code of SW-first algorithm

```
1  construct the BDD of the circuit;
2  compute the 0 and 1 probability for each signal;
3  //   traverse from the PO to PI to find control signals;
4  while (gate is not checked) do
5      finder (input1, 1, parity);
6      finder (input2, 2, parity);
7      for (gate i in block 1) do
8          if depth>maxd or parity %2≠1 or unavailable gate type(i) then
9              ctrl_signal_1(i) = 0;
10         else
11             ctrl_signal_1(i) = label_1[i];
12 recompute switching activities of fan-out cone of i;
13 SUM_sw_IN1(i) = Σ_all gates switching activity   ;
14 reduction_1(i) = SUM_sw_IN1(i) – SUM_sw;
15         end if
16     end for
17     for (gates in the block 2 ) do
18         same process as gate i;
19     end for
20     if reduction_1(i) > redeuction_2(i) and reduction_1(i) > 0 then
21        choose block 1 as controlled block;
22         SUM_sw = SUM_sw_IN1(i);
23         move to next gate;
24   end while
25
26 finder (node1, label, parity) {
27        if   node1 ≠  PI && have no branch then
28             label_1[node1] = label;
29             if   node1 is inverting gate then
30               parity++;
31             end if
32           finder (input_1, 2, parity);
33           finder (input_2, 1, parity);
34        end if
35        }
```

By checking the values of *reduction_1*(*i*) and *reduction_2*(i), the signal with more reduction of switching activity is selected as control signal. Meanwhile, the control signal should be given up if the reduction is less than 0, even though it can reduce more switching activity than the other one. Selected control signal and all logic gates that can be controlled by this signal are labeled to prevent to be traversed again in future procedure and all computation results about probabilities and switching activities executed in above control signal selecting process are remained as new circuit status.

One control signal assertion process is completed here and same procedure will be repeated to remaining parts of the circuit for clustering more controllable gates until all gates has been checked. Power savings obtained by using this algorithm is expected to be larger than the previous algorithm.

3.3.4 Maximum Depth Constraint

Proposed Pseudo PG could introduce delay overhead due to the control signal insertion. Thus a mechanism is introduced that inhibits the usage of some sleep signal if the delay larger than the original circuit. To implement this, depth from primary input D_{PI} and depth from primary output D_{PO} are computed first. Maximum depth $\mathrm{D_{max}}$ of original circuit is computed to be a criterion for depth constraint.

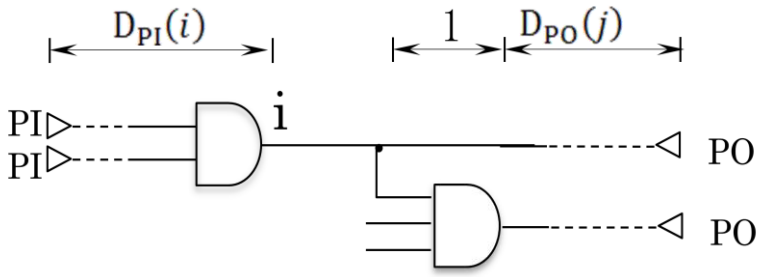


Figure 3-6: Depth variation

As shown in Figure 3-6, after assigning signal *i* as the control signal of gate *j*, new depth of circuit formed by this assignment can be computed by formula:

$$D_{new} = D_{\mathrm{PI}}(i) + D_{\mathrm{PO}}(j) + 1 \tag{3-4}$$

During application of clustering algorithm, before inserting one control signal, new depth of circuit by this possible assignment is compared with maximum depth D_{max}. The assignment can be given up if the D_{new} larger than D_{max} and vise versa.

Note that if $D_{PI}(i) > D_{PI}(j)$, which means that D_{PI} of the control signal larger than D_{PI} of the controlled gate, after control signal insertion, $D_{PI}(j)$ will be changed to $D_{PI}(i) + 1$ and D_{PI} of the all nodes in the transitive fan-out cone of the controlled gate should be recomputed. Similarly, if $D_{PO}(i) < D_{PO}(j)$, after inserting one control signal, the value of the $D_{PO}(i)$ will be change to $D_{PO}(j) + 1$ and it should also be recomputed the D_{PO} of the all nodes in the transitive fan-out cone of the controlled gate. With this mechanism, the proposed Pseudo PG can keep the maximum depth.

3.3.5 Glitch Avoidance Constraint

In general, switching activity in a circuit not only depends on the topologic structure and input patterns of the circuit, but also may vary with gate delay which introduces glitching transitions. Here, however, does not account for switching activity of glitches. The reason is that measuring the glitches requires an event-driven simulation and updating the amount of glitches with simulation iteratively is computationally too expensive.

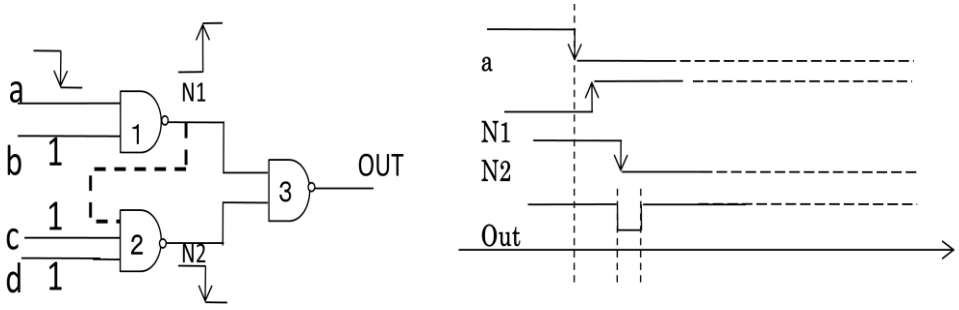


Figure 3-7: Occurrence of glitches

In practice, glitch power is taken into account during validation by power simulating. Several cases on the selection of controllable gates have been tested and analyzed. In some cases, some extra glitches are found at the gate gathering controlled gate and control signal, the power consumption of which is comparable with the power reduced by applying the Pseudo PG. Thus, the avoidance of glitches occurred by inserting control signal is necessary. The glitch is from the reconvergent fan-out like *e* in Figure 3-1. There are several paths from *e* to the gate 1, and a direct path includes 0 inverting gate (even number) and a path via the gate 2 includes one NAND gate (odd number). In the case, a glitch might happen at the output of the gate 4 depending on the delays of those paths. If the parity of the inverting gates number of all paths refers to even number, then there happens no glitch.

By erasing the paths with the odd number of inverting gates, the glitches can be avoided. Signal *e* should not be connected to the gate 2 for avoiding the glitch in Figure3-1. In the optimization, the parity should be considered when adding the control input. In Algorithm 1, line 9 and 31-33 describe parity checking for controlled gate. By adding this constraint, more power savings are expected.

3.4 Experimental Results

The proposed Pseudo PG method with clustering algorithm is implemented in C language and applied to ISCAS85 standard benchmark circuit. Experiments have been done on a computer with Intel Core 2 Duo CPU E8500 (3.16GHz) 3.25 GB of memory with Linux Ubuntu 8.04. At first, a benchmark circuit is read into BDD package, to separate the gates with several fan-in into two input gates, and determine the control signal and controllable gates by using the proposed algorithm. Then, probability of the output was calculated for all controlled gates after control signal insertion. For each gate in the circuit, the switching activity $SW(i) = 2 \cdot P(i) \cdot \big(1 - P(i)\big)$ is computed in the above procedure, where $P(i)$ is 1-propoability of gate i. The accumulation of the switching activity of all gates in the circuit is called as *SUM_sw*, and is used to evaluate the effect of the proposed clustering algorithm. The actual power consumption and propagation delay are simulated and the results are given in second part.

3.4.1 Switching Activity Reduction with Pseudo Power Gating

Table 3-1 to Table 3-2 shows the experimental results of the Pseudo PG method with basic algorithm, *pN*-algorithm and *SW*-first algorithm. In Table 3-1, column 2, 3, 4 show the switching activity accumulation of all gates, number of total gates and initial depth in the original benchmark circuits respectively. Column 5, 6 and 7 describe the number of power controlled gates and control signals, recomputed depth and switching activity accumulation after applying basic clustering algorithm. Table 3-2 shows the corresponding objects achieved by applying *pN*-first algorithm and *SW*-first algorithm.

Table 3-1 Results of basic algorithm.

Circuits	Initial circuits			Basic algorithm(*N*-based)		
	Initial *SUM_sw*	# of total gates	Initial. depth	# controlled gate/ # ctrl signals	Depth	*SUM_sw*
C432	83.29	252	31	134/26	45	80.83(2.95%)
C499	160.38	454	19	74/14	28	145.65(9.19%)
C880	142.11	435	30	181/63	41	110.44(22.29%)
C1355	213.85	590	26	178/118	41	209.87(1.86%)
C1908	370.69	1057	44	465/160	61	358.04(3.41%)
C2670	528.88	1400	40	621/191	59	502.28(5.03%)
C3540	650.16	1983	56	1115/289	83	570.47(12.26%)
C5315	1126.58	2973	52	1372/399	70	1090.92(3.17%)
Avg.						7.52%

From the experimental results, it is obtained that three types of clustering algorithm reduced total switching activity, and the *SW*-first algorithm gains the best reduction 21.67% on average. 7.52% of reduction is obtained by the basic (*N*-based) algorithm and the *pN*-based algorithm gains 5.94%. The *pN*-based algorithm is the best in the CV-based power gating, but the *N*-based algorithm is better than the *pN*-based algorithm in the proposed Pseudo PG method. Thus, the comparison between *SW*-based algorithm and *N*-based algorithm is mainly considered in following experiments. Note that, the increase of depth is not considered in above experiment. To test how the Pseudo PG affects the performance of the circuit, the circuit depth after inserting control signal is checked. The data show that the maximum depth of circuit elongated, ranging from 1.3 to 1.6 times (1.4 times on average) than that of original circuits by applying basic algorithm. For *SW*-first algorithm, a better reduction than

basic algorithm is achieved, even though the maximum depth only ranges from 1.1 to 1.5 times (1.3 times on average) than that of original circuits.

The proportions of the controlled gate of three algorithms are 41.50%, 26.51% and 23.77% respectively. These results indicate that the reduction of *SUM_sw* is not depending on the number of controlled gate. As mentioned in the subchapter 3.3.3, the actual switching activity variation should be considered as the parameter to monitor the efficiency of Pseudo PG. On the other hand, the more gates are controlled, the lager maximum depth of circuits can be leaded by inserting control signals to the gates on the critical path. The maximum depth constraint is urgently required to prevent circuit from significant performance degradation.

Table 3-2 Results of *pN*-based and *SW*-first algorithm.

Circuits	*pN*-based algorithm			*SW*-first algorithm		
	# controlled gate/# ctrl signals	Depth	*SUM_sw*	# controlled gate/# ctrl signals	Depth	*SUM_sw*
C432	82/16	40	74.60(10.44%)	43/21	36	70.59(15.25%)
C499	33/13	30	151.58(5.48%)	78/14	28	143.82(10.33%)
C880	167/132	35	131.54(7.44%)	114/42	31	80.35(43.46%)
C1355	35/33	30	212.99(0.40%)	106/46	36	196.63(8.05%)
C1908	294/121	51	354.27(4.43%)	233/73	48	271.10(26.87%)
C2670	382/111	45	500.64(5.34%)	377/80	59	336.68(36.34%)
C3540	853/284	67	613.17(5.56%)	691/228	66	530.18(18.45%)
C5315	887/366	68	1033.03(8.30%)	831/219	64	961.94(14.61%)
Avg.			5.94%			21.67%

Table 3-3 Comparison between basic and *SW*-first algorithms (under maximum depth constraint).

Circuits	Basic algorithm (*N*-based algorithm)		*SW*-first algorithm	
	# controlled gate/ # ctrl signals	*SUM_sw*	# controlled gate/ # ctrl signals	*SUM_sw*
C432	32/23	81.52(2.12%)	28/18	69.61 (16.43%)
C499	8/8	156.93(2.15%)	8/8	156.93 (2.15%)
C880	159/62	125.69(11.56%)	112/40	111.29(21.69%)
C1355	32/32	211.85(0.93%)	12/12	209.90 (1.85%)
C1908	386/138	362.52(2.20%)	215/67	274.28(26.01%)
C2670	520/180	505.77(4.37%)	305/94	343.16(35.12%)
C3540	953/182	580.18(10.76%)	576/127	530.18(18.45%)
C5315	1249/352	1085.93(3.61%)	803/206	961.10(14.69%)
Avg.		4.71%		17.05%

Afterward, steady maximum depth constraint is executed and the reductions of SUM_sw with both basic and SW-first algorithm are evaluated. The experimental results under the steady maximum depth constraint are presented in the Table 3-3. For the basic algorithm, the number of controlled gates is decreased comparing with the cases without depth constraint. The reduction of *SUM_sw* is decreased from 7.52% to 4.71% on average. On the other side, the proportion of controlled gate decrease from 20% to 14% with *SW*-first algorithm under the depth constraint. However, there is still 17% reduction of *SUM_sw* even under the constraint.

In the circuit C499 and C1355, the reduction of *SUM_sw* based on the *SW*-first algorithm is obviously smaller than other circuits. In these two circuits, XOR gates occupy half of the total gates, and they cannot be controlled by any

signal, because XOR gates have no controlling value. In C1355, even though XOR gates constructed with four 2-input NAND gates, only few gates can be controlled under the maximum depth constraint.

Experimental results in above 3 tables show the switching activity reduction of the circuit achieved by applying proposed Pseudo PG with different clustering algorithm. However, these results do not represent the end state and the final estimation of efficiency of Pseudo PG method. In addition, although the depth constraint is executed to the circuit and significant performance degradation is prevented, but actual propagation time is not be simulated. Therefore, the actual power and propagation delay is evaluated in the following subchapter.

3.4.2 Actual Power Reduction with Pseudo Power Gating

To check the actual effectiveness of the Pseudo PG method, the power consumption of these benchmark circuits is simulated with VDEC Rohm 0.18μm process technology on a computer with AMD Opteron 852 (2.8GHz×4, 32GB memory). After applying the Pseudo PG and using both basic algorithm and *SW*-first algorithm, the benchmark circuits are transformed into gate level Verilog description by using a converter program developed by us. The transformed netlist is layouted with Astro and parameter extraction is executed with Calibre. The power is evaluated by the nanosim using the netlist extracted from the layout results. 1000 random input patterns (0→1, 1→0 each occupy 25%) were put in to the simulation. Afterward, the wave of each signal can be seen in the wave view tool, Cosmoscope, to check the accuracy of logic after applying Pseudo PG.

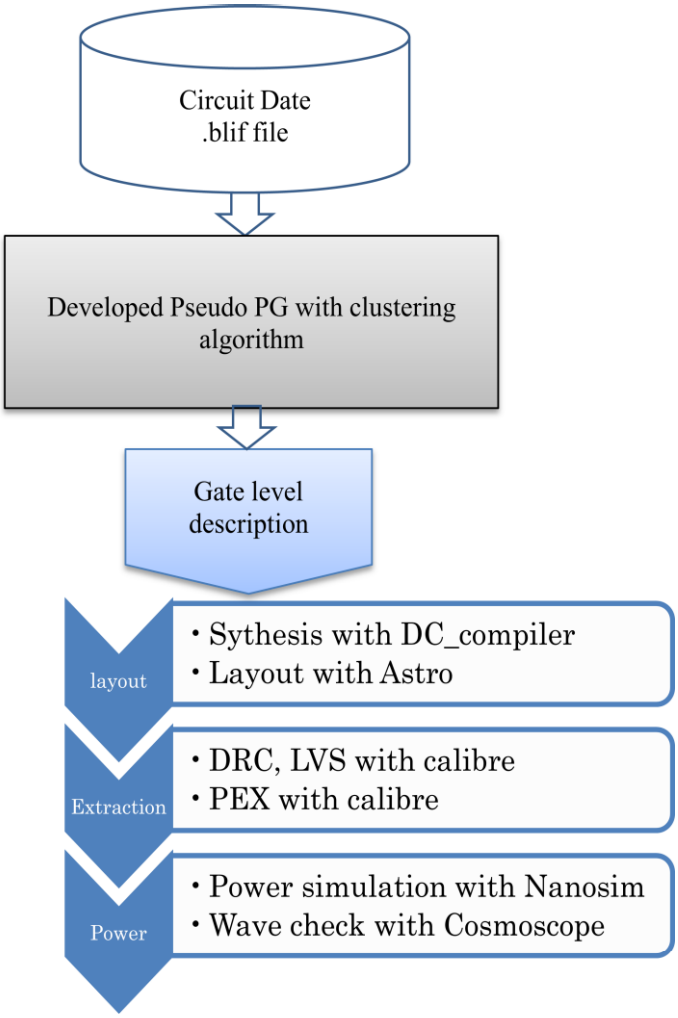


Figure 3-8: Power simulation flow

For checking the performance variation of the circuits caused by applying our method, the longest propagation delay is simulated after layout even the depth of the circuit is kept by the depth constraint. The simulation flow is shown in Figure 3-8. Table 3-4 shows the power consumption result of original circuit and the circuit applied Pseudo PG with basic algorithm and SW-first algorithm. Both algorithms are realized under the depth constraint.

Table 3-4 Power results with two algorithms (under maximum depth constraint).

Circuits		Basic algorithm (N-based algorithm)		SW-first algorithm	
	Initial Avg.power (uW)	Avg. power (uW)		Avg. power (uW)	
C432	156.59	152.33	2.7%	150.81	3.7%
C499	221.42	223.43	-0.9%	214.73	3.0%
C880	253.89	231.59	10.47%	206.92	18.5%
C1355	352.3	344.70	2.2%	332.74	5.5%
C1908	553.72	581.19	-4.9%	543.71	1.8%
C2670	816.35	747.43	8.4%	684.87	16.1%
C3540	924.74	876.67	5.22%	857.98	7.24%
C5315	1734.33	1522.63	12.2%	1469.55	15.2%
Avg. rate			4.42%		8.70%

It can be seen that the power reduction by applying the Pseudo PG with the basic algorithm is achieved by 4% on average under the maximum depth constraint, while 9% power reduction is obtained by the *SW*-first algorithm. As the results of switching activity reduction evaluated in subchapter 3.7.1, the SW-first algorithm reduces another 4% of power than the basic algorithm and proved our prospective of power saving efficiency.

Table 3-5 shows the maximum propagation delay result of original circuit and the circuit applied Pseudo PG with basic algorithm and SW-first algorithm. Both algorithms are realized under the maximum depth constraint to avoid depth increase.

Table 3-5 Delay simulation results with two algorithms (under maximum depth constraint).

Circuits		Basic algorithm (N-based algorithm)		SW-first algorithm	
	Initial Delay(ns)	Delay(ns)		Delay(ns)	
C432	2.75	2.88	-4.73%	2.77	-0.73%
C499	1.79	1.79	0.00%	1.79	0.00%
C880	2.25	2.59	-15.11%	2.35	-4.44%
C1355	1.95	1.91	2.05%	1.91	2.05%
C1908	2.71	3.55	-31.00%	2.71	0.00%
C2670	3.09	3.31	-7.12%	3.30	-6.80%
C3540	3.88	4.89	-26.03%	3.90	-0.52%
C5315	3.45	3.98	-15.36%	3.63	-5.22%
Avg. rate			-12.16%		

By adding the maximum depth constraint to basic clustering algorithm, the increase in longest delay of circuits ranging 0% to 30% (12% on average). Meanwhile, *SW*-first clustering algorithm increase delay by 0% to 6% (2% on average). For circuit C1355, the longest delay is even decreased in both clustering algorithms.

The power saving rate is smaller than the reduction rate of switching activity because the possible occurrence of glitches dose not taking into account to switching activities. Hence, another power simulation is executed with consider to avoid glitch caused by inserting control signal in the Pseudo PG. Table 4 shows the reduction *SUM_sw* and actual power simulation results of circuit with *SW*-first algorithm under the glitch avoiding constraint and maximum

depth constrain. For individual circuits, the switching activity reduction achieves 2%~14% (8% on average). In the other hand, power saving achieves 3%~25% (13% on average). By eliminating the glitch power, saving effect is larger than reduction in *SUM_sw*.

Table 3-6 Switching activities and power results with glitch avoiding constraint (under maximum depth constraint).

Circuits	# controlled gate/ # ctrl signals	*SUM_sw*(uW)	Avg. power(uW)	delay
C432	19/9	79.05(5.09%)	147.23(5.98%)	2.85(-3.64%)
C499	8/8	153.92(4.03%)	214.80(2.99%)	1.79(0%)
C880	99/27	123.98(12.76%)	206.63(18.61%)	2.37(-5.33%)
C1355	8/8	208.65(2.43%)	314.63(10.69%)	1.95(0%)
C1908	215/67	360.44(2.77%)	518.00(6.45%)	2.78(-2.58%)
C2670	269/70	474.26 (10.33%)	611.26(25.12%)	3.20(-3.56%)
C3540	667/139	557.62 (14.23%)	767.09(17.07%)	3.89(-0.26%)
C5315	584/229	1027.32(8.81%)	1469.54(15.27%)	3.63(-5.22%)
Avg.		7.56%	12.77%	-2.53%

Under both the maximum depth constraint and the glitch avoiding constraint, on average 4.12% switching activity reduction and 2.94 % actual power reduction is reached by applying the *pN*-based algorithm. The results indicate that, instead of the parameter N_E, total reduction of switching activity is more appropriate to represent the expectation of power savings in the proposed Pseudo PG method.

3.5 Summary

In this chapter, pseudo power gating (Pseudo PG) method is proposed to optimize the power consumption of combinational circuits by reducing the unnecessary switching activity of internal gates of circuits. The proposed method is based on the feature of controlling value which can decide an output of a gate with only one input. If one input of a gate takes the controlling value, the other inputs value become unnecessary. Thus, we can reduce the switching activity of the block that generates unnecessary inputs by inserting the input taking the controlling value as a control signal. Basic algorithm for selecting the control signal and controlling gate is introduced. This correspond to make a cluster of gates with respect to each control signal. In the basic algorithm, the candidate control signals are sorted with the number of controllable gates in corresponding cluster, and control signals are fixed one by one with the order of gate numbers

By analyzing the relation between switching activity change of controlled gates and controlling probability, switching activity first (*SW*-first) algorithm is proposed to maximize the reduction of switching activity in entire circuit. This algorithm provides optimal clustering of controlled gates and the control signal.

In order to get the maximum power reduction, a constraint is introduced to prevent the occurrence of glitches where a control signal can be connected to gates on the path with even number of inverters. Furthermore, considering the possible depth increasing at the insertion of control signal, the performance of circuit is maintained by using steady maximum depth constraint and a control signal is connected to gates only when the maximum depth of circuit does not change.

The proposed method has been implemented in C language and applied to ISCAS85 standard benchmark circuits based on the existing BDD manipulation system. Experimental results show that 1.0%~11.6% of reduction of switching activity is obtained by the basic size based algorithm and 1.9%~26.0% reduction is obtained by the SW-first algorithm. The power consumption is evaluated by simulating the netlist which is extracted from the layout using the VDEC (VLSI Design and Education Center in Tokyo University) Rohm 0.18μm process. Evaluations on the ISCAS'85 benchmarks show that the reduction of total power consumption has been achieved to 3.0%-25.3% (12.8% on average). This work provides a solution for switching activity reduction in very fine-grained region of logic level LSI design with small overhead.

Chapter 4

Transitive Fan-in based Algorithm for Further Power Saving

4.1 Introduction

In the previous chapter, Pseudo Power Gating (Pseudo PG) method is proposed based on the controlling value of a logic gate and is applied to designs with usual libraries and measured the power consumption. The leakage power reduction effect might be less, but the dynamic power reduction is more compared to the original controlling value based (CV-based) power gating method. Note that the original CV-based method is for power gating, and for Pseudo PG, several new features should be considered for Pseudo PG. Pseudo PG method can be seen as a kind of signal gating, but is much more fine-grained in area-direction and in time-direction and has smaller overhead compared with conventional signal gating methods.

In Pseudo PG method, one input of a logic gate is used as the control signal to blocks computing other inputs. The input is connected to each gate in

other blocks as an additional input. If the input takes the controlling value, it can stop the unnecessary switching activity of other blocks. The other blocks are called as power-controlled blocks. While the control signal takes non-controlling value, the power-controlled block work normally and the output of the focusing gate is determined by both inputs. The stopped switching activity depends on the power-controlled block and its control signal. To automatically apply the Pseudo PG method into combinational circuits and to detect conditions favorable for power saving, several clustering algorithms in Chapter 3.2-3.3 have been developed. In those clustering algorithms, switching activity first (*SW*-first) algorithm considering the actual variation of switching activities and obtained outstanding results in power reduction.

However, the maximum depth constraint might limit the potential power reduction is found in Pseudo PG method, since the clustering process gives up to control the target gate if the maximum depth of the circuit would change. Several gates might lose the opportunity to be controlled due to this constraint. In entire circuit, such power dissipation accounts for a considerable amount of overall power consumption.

In order to overcome the maximum depth constraint to obtain further power savings, a control signal with smaller depth should be found and an advanced clustering algorithm will be proposed in this chapter. The algorithm is based on the propagation property of a controlling value. An input value of a gate might be a controlling value of other fanout gate. In other words, transitive fan-ins of a gate might take a controlling value of the gate. The proposed algorithm uses such transitive fan-in property. An input with controlling value can become control signal to disable the power of blocks computing other inputs of focusing gate. In the transitive fan-in based algorithm, not only the direct input but also the transitive fan-ins are considered as control signal if the fan-in's

controlling value becomes the controlling value of the focusing gate. Transitive fan-ins have the smaller depth and might able to be a control signal to the blocks of other inputs.

The rest of this chapter is organized as follows: Section 4.2 describes control signal extension mechanism for advanced transitive fan-in based algorithm. Section 4.3 shows a method to reduce total number of the control signal with same power saving effectiveness. The experimental results are shown in the Section 4.4 and this chapter is summarized in the Section 4.5.

4.2 Transitive Fan-in based Clustering Algorithm

In Pseudo PG, an extra input might be assigned to several gates, as a control signal for each control signal, a cluster of gates can be defined for each signal. Since different assignments of control signal lead to different switching activity reduction, a clustering algorithm is important for selecting optimal control signal and corresponding power-controlled block to obtain the maximum power reduction. In the clustering, less switching activities are preferable to obtain more power saving. Hence, *SW*-first algorithm has been proposed in Section 3.3 with consideration of actual switching activity variation gained by the inserted control signal. The basic strategy of *SW*-first algorithm is to sort the power-controllable blocks according to the value of switching activity reduction and select the block with highest value as first power-controlled block and corresponding control signal are assigned to the block. Then the same procedure is applied to the next sort.

During the clustering, the maximum depth constraint was exerted and the gate that makes the circuit depth larger than the original one after control signal assertion would be excluded from the power-controlled block. In *SW*-first algorithm, about 40% gates were excluded by the depth constraint and the

expectable power reduction was limited. Hence, a new algorithm that could break through the restriction of the depth constraint is expected as the first improvement.

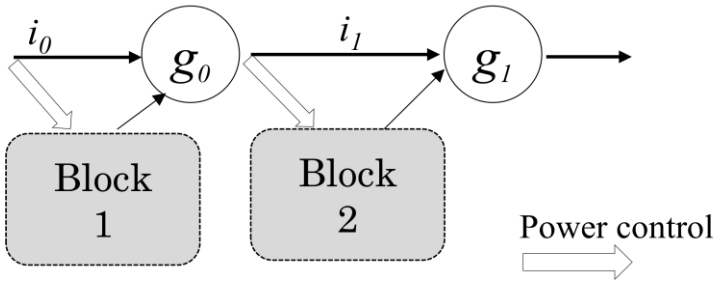
Section 3.4 addressed that new depth of a gate g after one assignment of a control signal i can be calculated by $D_{\mathrm{PI}}(i) + D_{\mathrm{PO}}(g) + 1$, where $D_{\mathrm{PI}}(i)$ is the depth of signal i from primary inputs and $D_{\mathrm{PO}}(g)$ is the depth of gate g from primary outputs. From the point of view of the depth, the signal closer to the primary inputs is better as a control signal, because it will make $D_{\mathrm{PI}}(i)$ small and new depth will not exceed the maximum depth. Thus, it is expected to find a control signal not only in the direct input but also in the transitive fan-in. In other words, a candidate control signal might control the gate blocks to the transitive fanout.

For the further discussions about propagation of controlling conditions, several definitions are given first. Assume that signal i_0 is one input of gate g_0, and a series of gates cascading g_0 form Path $P = \{g_0, g_1, \ldots, g_n\}$.

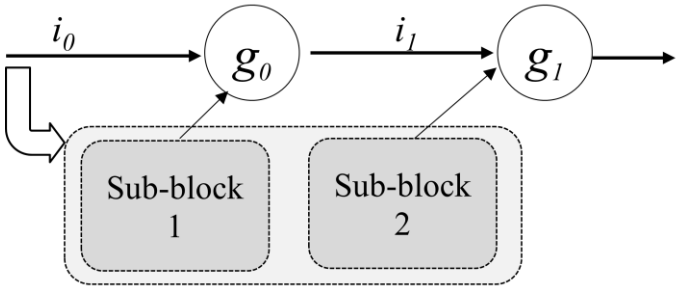
Definition 1: a path P is a **CV-propagation path** of a signal i_0, if the output of g_i ($i = 0,1,\cdots,n$) $\in P$ is determined when i_0 takes the controlling value.

Definition 2**:** a **side-input** to gate g_i along P is any input other than g_{i-1}.

If one signal takes controlling value, the output of each gate along the associated CV-propagation path can be decided. Therefore, the computation of the block generating side-inputs becomes unnecessary. In other words, a signal can control the part generating the side-input to the gates along the CV-propagation path. This property is illustrated by using an example in Figure 4-1. If both gates g_0 and g_1 in Figure 4-1 a), which are connected in series, are AND gates, then i_0 can control not only gate g_0 but also gate g_1. So, a path $\{g_0, g_1\}$ is a CV-propagation path of signal i_0.



a) Previous *SW*-first clustering scheme.



b) Transitive fan-in based clustering scheme.

Figure 4-1: Improvement of clustering algorithm

If the previous *SW*-first clustering algorithm is applied to the circuit in Figure 4-1 a), signal i_0 and i_1 are selected as control signal of block 1 and 2 respectively, when they do not violate the depth constraint. In the original *SW*-first clustering algorithm, the direct inputs are considered. Whereas, in the advanced clustering algorithm of in this research, the CV-propagation path is considered. If a path $\{g_0, g_1\}$ is a CV-propagation path of signal i_0, then i_0 can control block 2. Note that i_0 is a transitive fan-in of g_1, and in the advanced algorithm not only the direct input but also the transitive fan-ins are considered. So block 1 and 2 can be controlled by i_0 as shown in Figure 4-1 b). Since the depth of i_0 is smaller than that of i_1, the maximum depth constraints can be satisfied easier than i_1.

The advanced clustering algorithm is expected to increase the chance to

find the power control which might lose in the previous clustering algorithm because of the maximum depth constraint. At the same time, the algorithm is suitable to dual-stage control. For example, sub-block 2 can be controlled by i_1 and i_0 as a dual-stage manner, which improve the control capability. The details will be discussed in Section 4.

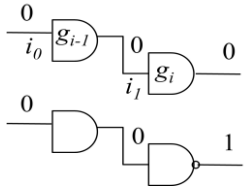
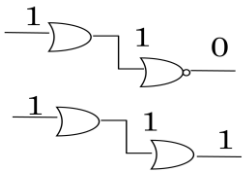
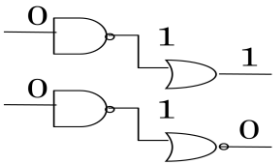
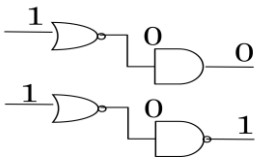
The control signal selection strategy of advanced algorithm is the same with existing *SW*-first algorithm, which selects the control signal from the one forming highest reduction of switching activities. The difference is that the power-controlled block for one control signal is extended according to the property of CV-propagation path. Since each signal in the circuit can be potential control signal, CV-propagation paths of each signal should be confirmed first. Here, another definition is shown for more explanation.

Definition 3*:** a signal (g_{i-1}, g_i) denoting an edge between gates g_{i-1} and g_i is a ***receiver of a signal (g_{i-2}, g_{i-1}), if the controlling value of g_i is the same as the output value of g_{i-1} decided by the controlling value of g_{i-1}.

The above definition shows that the propagation property of controlling value. The controlling value of g_{i-1} decides its output, and the output value also becomes the controlling value of g_i and decides its output. Simply put, on a CV-propagation path, the output of one gate is the *receiver* of the input which belongs to this path. For example, signal i_1 is a *receiver* of signal i_0 in Figure 4-1, since signal i_0 with g_0's controlling value can decide the output of g_1.

There are several cases on the receiver structure. Table 4-1 shows all the possible gate type combinations that allow the input signal to have *receiver*. According to Table 1, we check the each transitive fanout to find out all *receivers* and figure out the CV-propagation paths of each candidate control signal.

Table 4-1 The gate type satisfying *definition 3* (i_1 is *receiver* of i_0)

g_{i-1}'s CV=0	g_{i-1}'s CV=1
	
	

For a signal, the CV-propagation path is not unique and the number related to the fanouts number. As shown in Figure 4-2, all CV-propagation paths of one signal can be formed by iteratively adding the *receiver* until reaching one gate that does not satisfy *Definition 3*.

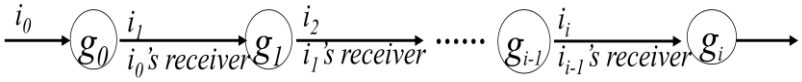


Figure 4-2: CV-propagation path of the signal i_0

Figure 4-3 shows an example actual example of CV-propagation path exist in benchmark circuit C1355. Signal 'GAT_329' has 11 paths satisfying definition of CV-propagation path. So for the candidate control signal 'GAT_329', the side inputs to the 11 paths can be clustered as on block if effective. So far, finding *receiver*s for each candidate signal and clustering of gate blocks are finished and the method of deciding control signal will be shown in following.

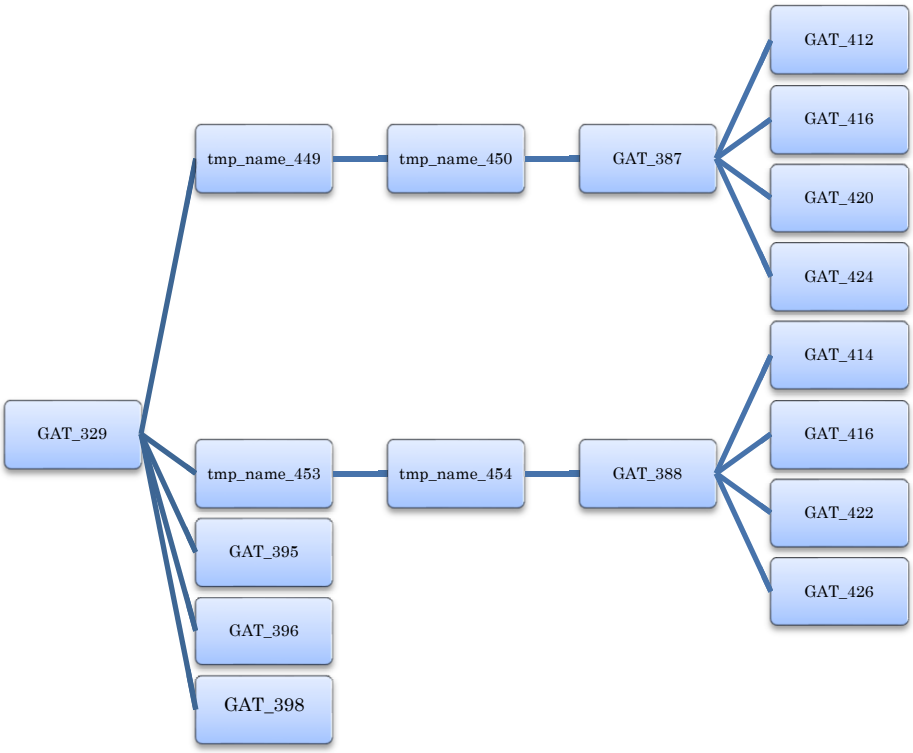


Figure 4-3: 11 CV-propagation path of GAT_329 in Circuit C1355

For any candidate control signal *i*, a backward traversal procedure is executed to traverse the part generating side inputs, and controllable gates of this part can be decided. Then, the switching activity reduction formed by control signal *i* is accumulated as one value: *sw_red*[*i*]. Afterwards, the same procedure is executed on all *receiver* of the signal *i*, and searched controllable gates of all *receivers* are merged to the power-controlled block of signal *i*. Reductions of switching activities of these gates are added to *sw_red*[*i*]. Recursive procedure is executed repeatedly until no more *receivers* can be traversed. So far, final *sw_red*[*i*] of signal *i* is obtained and traversal program goes to next candidate control signal.

Algorithm2 Pseudo code of transitive fan-in based algorithm

```
 1  Construct the BDD of the circuit;
 2  Compute the 0 and 1 probability for all signals;
 3  Construct receiver for all signals;
 4  while (number of signals to check > 0) do
 5    for all control signal candidates do
 6      COMPUTE_SW(candidate, 0.0);
 7    end for
 8    Select the signal k with top value of sw_red;
 9    Assign k to block k as control signal;
10    Renew BDD;
11  end while
12
13  COMPUTE_SW(signal i, sw_red( i ) ) {
14    BACKTRACE(side-input of i);
15    Add controllable gates to block i;
16    Compute sw_red( i );
17    for all receiver of i do
18      COMPUTE_SW(receiver, sw_red( i ) );
19    end for
20  }
21
22  BACKTRACE( gate1 ){
23    if gate1≠ PI and have no branch then
24      label_1[gate1] = label;
25      feedback1 = BACKTRACE(input_1 );
26      feedback2 = BACKTRACE(input_2 );
27     if feedback1 = 1 and feedback2 = 1 then
28       gate1is excluded from Block;
29       return 0;
30       else return 1;
31     end if
32    end if
33  }
```

Finally, the candidate signal which can achieve maximum value of *sw_red* is selected as the control signal, and the assignment for control signal and controllable gates clustered in the above traversal program is realized. One control signal assignment is completed here and the same process will repeat to all the remaining signals and gates for several times until *sw_red* is less than 0. Based on the above description, the outline of our transitive fan-in based clustering algorithm is represented in **Algorithm 2**.

4.3 Control Signal Reduction of Power-controlled Block

Since transistors increase by the control signal insertion, it is desirable to get maximum power reduction by connecting minimum number of control signals. If all input blocks of one gate are controlled, no transition event will occur when the control signal takes controlling value. Therefore, the control signal to the gate is unnecessary, and the power can be reduced by disconnecting the control signal.

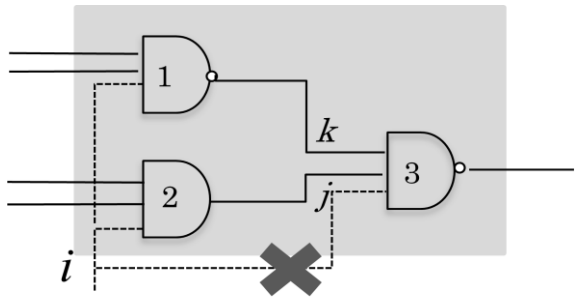


Figure 4-4: Control signal saving mechanism

Figure 4-4 shows an example. Let these 3 gates be controlled by the signal *i*. In this situation, the control signal *i* to gate 3 can be removed because both input blocks k and j are already under the control of the signal *i*. This is the effect of the transitive control by the control signal. Such control signal removal

has no effect to the switching activity reduction. Since these gates whose control signal is removed are still covered by the control signal, they should be marked as controlled gate to prevent becoming a candidate for control signal or power-controlled gate again in next procedure. The lines 25 to 28 in **Algorithm 2** correspond to the reducing control signal mechanism for power-controlled block.

Another point effecting evaluation of switching activity is number of transistor. Since the power saving is estimated in terms of amount of switching activity reduction, the switching activity of each gate is better to be represented with high reliability. Due to the different number of transistor in each gate, the amount of switching activities of entire circuit need to be accumulated in proportion by the number of transistors in individual gate. For example, the proportion coefficient of NAND is 1 while that of AND gate is 1.5 since two gates consist of 4 and 6 transistors respectively. This coefficient can be used to each gate to approach the realistic value of switching activity. Therefore, the number of transistors is also considered when we evaluate the switching activity reductions.

4.4 Experimental results

The newly proposed transitive fan-in based algorithm is implemented in C language and applied to a set of ISCAS85 standard benchmark circuits in the environment of BDD package. The switching activity of the circuit before and after implementing proposed approaches is calculated based on the BDD based computation. In order to evaluate actual power saving in the new proposed approaches and compare with previous research, the power consumption of these benchmark circuits is evaluated with the same CMOS process (VDEC Rohm 0.18μm process technology) and with the same experimental environments in Chapter 3. The power consumption is simulated by Nanosim and the function

correctness of circuit after applying the Pseudo PG is check by the signal wave checking with Cosmoscope. Experiments have been implemented on a computer with Intel Core 2 Duo CPU E8500 (3.16GHz) 3.25 GB of memory with Linux Ubuntu 8.04.

The experiments are divided into two parts. In the first part, we evaluate the switching activity reduction and actual power consumption after implementing transitive fan-in based clustering algorithm and compare the effectiveness with previous *SW*-first algorithm. After that, the propagation delay, area and the leakage power of the circuits applied proposed approaches is evaluated to check the performance degradation.

4.4.1 Power Reduction of Pseudo PG with Transitive Fan-in based Clustering Algorithm

For comparing with *SW*-first clustering algorithm as an existing single stage clustering algorithm, we implemented transitive fan-in based clustering algorithm on the same benchmarks. Table 4-2 and Table 4-3 show the comparison of two algorithms for Pseudo PG under the maximum depth constraint and glitch avoiding constraint.

Column 2 in Table 4-2 displays the total number of gates in circuit and columns 4 to 7 are existing experimental results shown in Chapter 3.6 for previous *SW*-first clustering algorithm. They respectively present the number of gates in the power-controlled blocks, power consumption and power reduction rate evaluated after running the *SW*-first algorithm as well as the algorithm execution time.

Table 4-2 Results of *SW*-first algorithm

Circuits	Total gates *Nc*	Controlled gates *Ns*	*Red. %of SUM_sw*	Run time(s)
C432	252	19	5.09%	0.32
C499	454	8	4.03%	0.36
C880	435	99	12.76%	0.53
C1355	590	8	2.43%	0.35
C1908	1057	215	2.77%	0.41
C2670	1400	269	10.33%	3.54
C3540	1983	667	14.23%	7.90
C5315	2973	584	8.81%	33.36
Avg.			7.56%	

Table 4-3 Results of transitive fan-in based algorithm

Circuits	Total gates *Nc*	Controlled gates *Ns1*/covered gates *Ne*	*Red. %of SUM_sw*	Run time(s)
C432	252	34/4	12.47%	0.50
C499	454	40/0	4.53%	0.30
C880	435	121/33	26.61%	0.50
C1355	590	16/0	2.84%	0.30
C1908	1057	115/45	8.68%	0.40
C2670	1400	303/31	16.15%	3.60
C3540	1983	473/108	22.96%	7.40
C5315	2973	887/142	20.90%	31.00
Avg.			14.39%	

Columns 2 to 5 in Table 4-3 present the corresponding contents obtained by applying transitive fan-in based algorithm. Column 3 additionally shows the number of covered gates, which include directly controlled gates as well as the indirectly controlled gates, whose inputs are already controlled by same signal, as discussed in Subsection 3.3. The number of affected gates (total number of controlled gates and covered gates) and control signal is significantly increased in the new transitive fan-in based clustering algorithm even under the depth constraint. For individual circuits, the rates of affected gates range from 2.71% to 35.40% (20.61% on average) of total gates in the circuit. The considerable reason is that the controllable gates excluded by the depth constraint in *SW*-first algorithm are selectively kept in transitive fan-in based algorithm.

The further reduction of switching activities is obtained by transitive fan-in based algorithm. Reduction of switching activities ranges from 2.84% to 26.61% (14.39% on average) in benchmark circuits. At present, the proposed Pseudo PG method is applied to circuits up to 2973 gates. The execution time is about 45seconds for the largest one. When applying to large circuits, we should divide the circuit into several parts. In the case, the result is not optimum, but the power might be reduced using the local controlling relation in the divided part.

To check the actual power reduction efficiency of proposed transitive fan-in based algorithm, the power consumption of these benchmark circuits is simulated with same experimental environment of previous Section 3.7 which presented actual power results of Pseudo PG with *SW*-first algorithm.

Table 4-4 Power reduction of transitive fan-in based algorithm

Circuits	Initial power Pc(μW)	Avg. power Ps(μW) (red. %)	Red.
C432	156.59	125.61	19.78%
C499	221.42	213.41	3.62%
C880	253.89	198.55	21.80%
C1355	352.3	284.84	19.15%
C1908	553.72	435.23	21.40%
C2670	816.35	590.08	27.72%
C3540	924.74	730.95	20.98%
C5315	1734.33	1422.98	17.95%
Avg.			19.05%

After applying the Pseudo PG using transitive fan-in based algorithm, the benchmark circuits are transformed into gate level Verilog description by using a converter program developed by us. The transformed netlist is layouted with Astro and parameter extraction is executed with Calibre. The power is evaluated by the nanosim using the netlist extracted from the layout results. 1000 random input patterns (0→1, 1→0 each occupy 25%) were put in to the simulation. Afterward, the wave of each signal can be seen in the wave view tool, Cosmoscope, to check the accuracy of logic after applying Pseudo PG.

Column 3 in Table 4-4 shows the simulation results of average power consumption. Power reduction achieved by the transitive fan-in based algorithm ranges from 3.62% to 27.72% (19.05% on average) and exceeding the previous SW-first algorithm by 6.28% and outperforms the other clustering algorithm studied so far. It can be seen that transitive fan-in based algorithm shows obvious advantages in terms of both accuracy and efficiency.

4.4.2 Propagation Delay, Area, Leakage Power Analysis

In this experiment, although the maximum depth constraint is exerted to avoid extreme degradation of timing performance in Pseudo PG method, the increase of propagation time and area overhead are unavoidable. To check the timing performance and area variation of the circuits before and after applying the Pseudo PG method with proposed approaches, the propagation delay and area of the circuits is evaluated by simulating the netlist after layout. SDF (Standard Delay Format) file is used to check propagation delay of longest path in the circuit. The SDF file stores the timing data generated by EDA tools for use at any stage in the design process. The data in the SDF file is represented in a tool-independent way so that we can use timing analyzing tool in the Design Compiler to check the timing information. Table 4-5 shows initial propagation delay and that of the *SW*-first algorithm while Table 4-6 presents the performance degradation caused by proposed transitive fan-in based algorithm.

Table 4-5 Propagation delay of SW-first algorithm

Circuits	Initial delay *Dc*(ns)	SW-first algorithm *D1*(ns)	Red. *(Dc-D1)/ Dc*
C432	2.75	2.85	3.64%
C499	1.79	1.79	0%
C880	2.25	2.37	5.33%
C1355	1.95	1.95	0%
C1908	2.71	2.78	2.58%
C2670	3.09	3.20	3.56%
C3540	3.88	3.89	0.26%
C5315	3.45	3.63	5.22%
Avg.			2.53%

Table 4-6 Propagation delay of transitive fan-in based algorithm

Circuits	Initial delay *Dc*(ns)	transitive fan-in based algorithm *D1*(ns) (red.)	Red. *Dc-D1*/ *Dc*
C432	2.75	2.85	3.64%
C499	1.79	1.79	0%
C880	2.25	2.40	6.67%
C1355	1.95	1.95	0%
C1908	2.71	2.82	4.06%
C2670	3.09	3.18	2.91%
C3540	3.88	4.01	3.35%
C5315	3.45	3.69	6.96%
Avg.			3.45%

Results indicate that the newly proposed transitive fan-in based algorithm lead to 0% ~ 6.67% (3.45% on average) of increase of propagation delay. Compared with SW-first algorithm, another 0.92% of degradation is resulted. Considerable reason is that increased number of power-controlled gates and corresponding additional inputs due to the extension of control signal with transitive fan-in based algorithm. For circuit C499 and C1355, both algorithm are not result in any performance degradation due to the small amount of power-controlled gate. As mentioned in the previous Chapter 3.7, this two circuits mainly consist XOR gates or XOR logic which cannot be controlled by any signal, therefore the proportion of power-controlled gates is obviously small than other circuits.

In addition, to evaluate the area overhead of the Pseudo PG method

with proposed transitive fan-in based algorithm, the area information after layout is also checked and presented in Table 4-7.

Table 4-7 Area overhead of transitive fan-in based algorithm

Circuits	Initial Area ($\times 10^3 \mu m^2$)	transitive fan-in based algorithm ($\times 10^3 \mu m^2$)	Inc.
C432	3.08	3.51	13.96%
C499	4.70	5.21	10.85%
C880	5.71	6.31	10.51%
C1355	7.38	7.46	1.08%
C1908	11.43	11.81	3.32%
C2670	16.24	16.82	3.57%
C3540	23.57	25.89	9.84%
C5315	36.60	41.43	13.20%
Avg.			6.29%

Results in Table 4-7 shows that, the proposed controlling path based algorithm results in 6.28% (on average) of area overhead.

Furthermore, the leakage power consumption is also measured for benchmark circuits and the experimental results are shown in Table 4-8. Although the leakage power in 0.18μm process only occupies negligible magnitude, it is decreased in most of the circuits after applying the proposed algorithm. Considerable reason of the leakage power reduction is the transistor stacking effect which occurred by inserting control signal to power-controlled gates.

Table 4-8 Leakage power of proposed algorithm

Circuits	Initial Leak. Power (nW)	path-based (nW)	Red.
C432	1.72	1.57	8.72%
C499	4.45	4.47	-0.45%
C880	0.50	0.43	14.00%
C1355	4.82	3.55	26.35%
C1908	1.93	1.50	22.28%
C2670	1.42	1.22	14.08%
C3540	3.17	0.90	71.61%
C5315	120.62	121.28	-0.55%
Avg.			19.51%

These overheads are moderate and the proposed transitive fan-in based algorithm is useful. The concept of transitive fan-in based clustering algorithm, which overcome the restrict depth constraint to extent the region of control signal to fan-in's area, also can be applied in the power gating method based on the controlling value.

4.5 Summary

In this chapter, an advanced transitive fan-in based clustering algorithm has been proposed to optimize control signal selection in pseudo power gating method which proposed in Chapter 3 to reduce the power dissipation of LSI circuits. The proposed approach includes two features: first, the transitive fan-in based clustering algorithm is proposed to improve potential power reduction limited by maximum depth constraint. Since the maximum depth constraint may

exclude several gates from power-controlled block, considerable potential power reduction is neglected. In the advanced clustering algorithm, the power-controlled block is extended by merging other blocks along the CV-propagation path of control signal in single-stage. Thus, the gates in the circuit, especially for the ones excluded by depth constraint, are given more opportunities to be efficiently controlled.

As the other feature, the power-controlled block is shrunken by removing the redundant connection of control signal. Partial gates, which only have already controlled input signals, are excluded from power-controlled block. By doing that, the penalty of control signal is decreased while keeping same power saving effect.

The experimental results show that 19.05% average power reduction has been achieved by applying proposed transitive fan-in based clustering algorithm. About 6% of power is saved further than previous SW-first algorithm with maximum depth constraint. The rates of affected gates range from 2.71% to 35.40% (20.61% on average) of total gates in the circuit. Power saving efficiency is outperforms other clustering algorithms studied so far with acceptable delay, area overheads. The considerable reason is that the controllable gates excluded by the depth constraint in *SW*-first algorithm are selectively kept in transitive fan-in based algorithm.

Chapter 5

Dual-stage Pseudo Power Gating

5.1 Introduction

In the previous chapter, a transitive fan-in based clustering algorithm is proposed to optimize the selection of control signal and corresponding power-controlled block in the Pseudo PG method. In this newly proposed algorithm, transitive fan-ins are considered as a control signal to care about the restriction of the steady maximum depth constraint. By merging several power-controlled blocks as one block through unifying control signal with smaller depth from primary input than original control signal, the chance of control is increased effectively.

However, the control signal might takes non-controlling value. No matter how effectively clustering algorithm is to be used, there still might exist unnecessary computations in power-controlled blocks, especially in large clustered blocks.

In this chapter, a novel dual-stage Pseudo PG structure will be proposed to cope with the demands of further power reduction. In the dual-stage Pseudo

PG, up to two control signals can be used for deactivating unnecessary computations, and an extra control is assigned inside Pseudo PG blocks. The transitive fan-in based clustering algorithm is suitable to dual-stage Pseudo PG structure for saving unnecessary power consumption while the existing control signal takes non-controlling value.

The rest of this chapter is organized as follows: Section 5.2 describes new structure of Pseudo PG for further power saving. Section 5.3 shows the specific realization mechanism of Pseudo PG to deal with evolved dual stage structure. The experimental results are shown in Section 5.4 and this chapter is summarized in Section 5.5.

5.2 Dual-stage Pseudo PG Structure

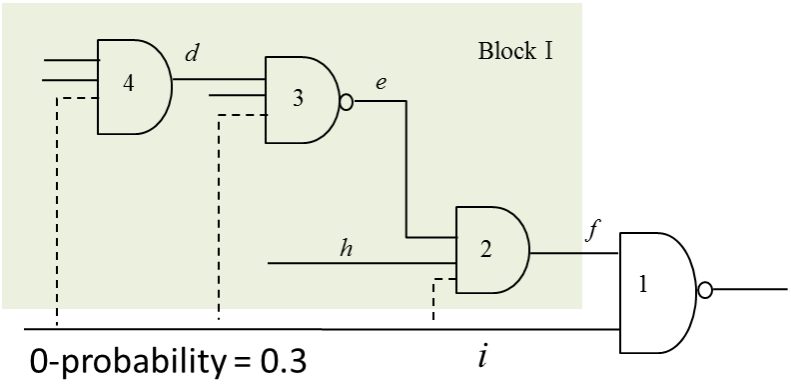


Figure 5-1: Application example of Pseudo PG

(part of circuit in Figure 3-1)

In this section, dual-stage Pseudo PG structure is shown. In dual-stage PG, two signals are used to control one gate. At first, we start from a single stage Pseudo PG structure and show how to find a control signal for dual stage.

Figure 5-1 shows an application example of Pseudo PG. In this figure the control signal *i* controls the power of gates 2, 3 and 4. Signal *i* is inserted to

these gates as an additional input. Three gates 2, 3 and 4 are power-controlled during signal *i* takes controlling value logic-0 with respect to the gate 1. If the 1-probability of signal *i* is 0.7, then these three gates has 0.3 of probability to be controlled by signal *i*. On the other hand, these gates are working normally in the rest of the time. It means that, when *i* takes the value-1 with 0.7 of probability, some unproductive switching activities might happen on these three gates. Hence, a new structure that could save further unnecessary switching activities in the existing power-controlled block is desired as the improvement of the Pseudo PG. In this following, an advanced Pseudo PG structure is proposed to cope with above demands.

The name "dual-stage" indicates that there are two stages in Pseudo PG control. By applying second stage Pseudo PG in the power-controlled block clustered from first stage Pseudo PG, more power reduction has been achieved.

To clarify the dual-stage Pseudo PG structure, the circuit in Figure 5-1 is used, where gates 2, 3 and 4 are controlled by the signal *i* with respect to the gate *1*. If the control signal *i* takes non-controlling value 1, the power-controlled block operates out of control. However if *h* takes the controlling value 0, then activities on gates 3 and 4 are unnecessary. An intuitive way to stop the unnecessary computation is to control the gate 3 and 4 by using signal *h*. As a result, the dual-stage structure is established by inserting second control signal to gate 3 and 4. The second stage control is shown as bold line in Figure 5-2.

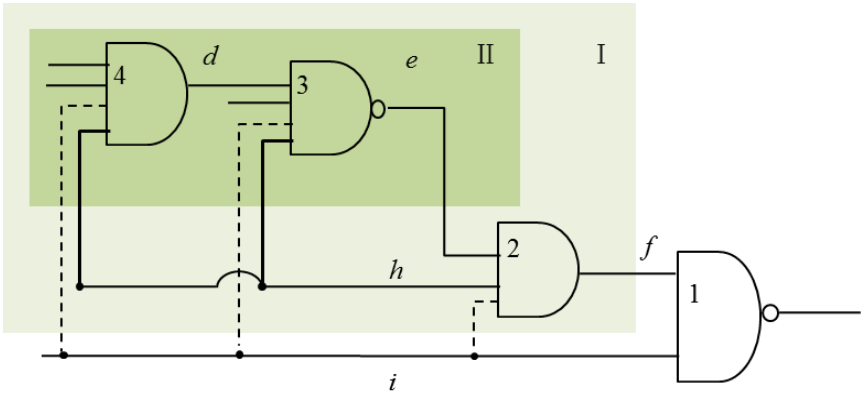


Figure 5-2: Dual-stage Pseudo PG structure for power-controlled block.

The dual-stage structure can improve the stopping capability compared to the single-stage structure. However, in the dual-stage Pseudo PG controlling structure the controlled gates require two additional inputs to receive the two control signals. Consequently, the area and depth increase should be considered when selecting two stage control signals and corresponding power-controlled blocks (clusters). Combined with proposed advanced clustering algorithm, the realization process of dual-stage Pseudo PG structure will be elaborated as follows.

5.3 Realization of Second Stage Pseudo PG

In Chapter 4, a transitive fan-in based clustering algorithm is proposed which can be used not only in single-stage but also in dual-stage Pseudo PG structure. The transitive fan-in based clustering algorithm enlarges the size of power-controlled blocks. In single-stage Pseudo PG, the larger size of the power-controlled block can be constructed but such block potentially contains the more unnecessary switching activities. In the transitive fan-in based algorithm, dual-stage structure is more aggressively desired. Therefore, for power saving effectiveness reasons, transitive fan-in based algorithm are applied

to dual-stage structure. The following part will show the dual-stage Pseudo PG realization with transitive fan-in based algorithm.

The procedure of dual stage structure realization using advanced clustering algorithm is shown in Figure 5-3. It is divided into two parts: first stage clustering and second stage clustering. To increase the efficiency of clustering algorithm for dual stage structure and to reduce the redundant traversing for searching control signal, two phase algorithm is introduced.

First stage	Transitive fan-in based algorithm is applied; *j* seleted as a control signal;
Second stage	For each power-controlled block j apply Dual_stage (power-contolled block[*j*]); Dual_stage (block){ Case 1: nodes in the Sub-block1 in Figure 5-4 if TOP_*red_sw*[*k*]>0, then *k* is control signal; Case 2: nodes located in Sub-block2 in Figure 5-4 **if** *red_sw*[*i*]>0 **and** *i* located on CV-propagation path, then *i* is control signal; }

Figure 5-3: Second stage realization

The first stage control signals and corresponding power-controlled blocks are obtained by applying the proposed clustering algorithm. Then, we regard the power-controlled block clustered in the first stage Pseudo PG as a new objective circuit for searching the secondly-controllable gates and corresponding control signal.

Second stage realization procedure is shown in Figure 5-3 and an application example is in Figure 5-4. The second stage clustering adopts almost the same procedure applied in the first stage. In the second stage clustering, objective circuits can be divided into two parts. One is the sub-block generating side input to the first gate on the CV-propagation path of the corresponding control signal, such as sub-block 1 in Figure 5-4. In this sub-block, the same clustering method as one in the first stage can be applied. Selected control signal is inserted to the gates which clustered to do the second stage control as second additional input.

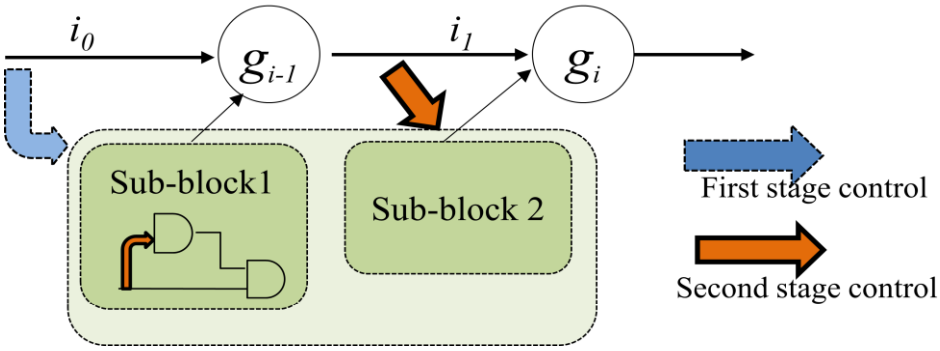


Figure 5-4: Second stage control signal selection.

On the other hand, in remaining sub-blocks, which generate the side input of the gates except first gate on the CV-propagation path, such as sub-block 2 in Figure 5-4, the second stage control signal can be selected directly: the signal on the CV-propagation path performs as the second stage control signal of the remaining parts. For example, signal i_1 can control the

sub-block 2 in the second stage.

This is from the following properties.

1. While the first stage control signal i_0 takes controlling value, sub-block 2 and gate g_i are under the control of i_0 without violating the functionality of g_i.
2. While the first stage control signal i_0 takes non-controlling value, the unnecessary computation in sub-block 2 also can be stopped by signal i_1 if it takes the controlling value of gate g_i in second stage. The way of searching control signals in the second stage is also suitable for the other blocks.

Compared with the case of sub-block 1, more gates can be obtained under the second stage by clustering as the case of sub-block 2. In practice, by marking the second stage control signal in the first stage clustering described above, the second stage clustering is efficiently realized without another recursive procedure, because the comparison and selection for gates of the sub-block have been done in the recursive traversing procedure of the first stage.

5.4 Depth Constraint and Glitch Avoidance

The maximum depth constraint is also necessary to maintain the timing performance of dual-stage Pseudo PG. In the second stage clustering, we use the virtual depth of each gate in which a control signal is inserted. If a control signal is inserted, then we recomputed the depth of a gate using the following formula:

$$D_{Gi} = \max\{D_{Gi},\ D_1 + 1,\ D_2 + 1\} \qquad (5\text{-}1)$$

where D_{Gi} represents the depth of the power-controlled gate, and D_1 is the depth of the first stage control signal. D_2 is the depth of the second stage control signal. When deciding control signal, the depth constraint should be considered by

calculating the virtual depth with formula (1). The virtual delay after inserting the control signals of two stages can be kept the same with the original circuit.

In Pseudo PG method, inserting control signal may result in occurrence of glitches under certain conditions as shown in Chapter 3.3.5. Previous solution is to remove control signal from the gate if the propagation path from this gate to the reconverged gate with the control signal has odd number of inverting gates such as NAND gate. In the proposed transitive fan-in based clustering algorithm of this research, the glitch avoidance method is almost the same as the previous solution. The difference is that the number of propagation paths might become two in the proposed dual stage method. So only if the total numbers of inverting gates in two paths are even, the signal *i* is added as the control to the controlled gate.

By the above ways, the performance degradation of the circuit can be avoided.

5.5 Experimental Results

The proposed dual-stage Pseudo PG is realized with transitive fan-in based algorithm, applied to the set of ISCAS85 standard benchmark. In order to evaluate actual power saving in the new proposed approaches and compare with previous research, the power consumption of these benchmark circuits is evaluated with the CMOS process (VDEC Rohm 0.18μm process technology). Experiments have been implemented on a computer with Intel Core 2 Duo CPU E8500 (3.16GHz) 3.25 GB of memory with Linux Ubuntu 8.04.

Experiment is divided into two parts. First, evaluation of the dual structure in the *SW*-first algorithm is accomplished. In fact, we can speculate that dual structure should have better power saving effect with transitive fan-in

based clustering algorithm than *SW*-first algorithm due to the enlarged power-controlled blocks. To check the efficiency of dual structure with both clustering algorithm, the switching activity reduction and average power reduction are evaluated.

Another one is the propagation delay, area and the leakage power of the circuits applied proposed approaches is evaluated to check the performance degradation with dual stage structure.

5.5.1 Dual-stage Pseudo PG with Transitive Fan-in based Clustering Algorithm

After the first stage Pseudo PG is realized by the transitive fan-in based algorithm, we added the second stage structure to the benchmark circuit. Table 5-1 gives the experimental results. Column 2 shows the number of controlled gates in the first stage of dual-stage Pseudo PG, which is the same as the results of transitive fan-in based clustering algorithm with single-stage Pseudo PG. The number of controlled gates in the second stage is shown in column 3. On average, 37.08% of gates in the power-controlled block clustered in first stage are further controlled by second stage clustering. The runtime of dual stage structure with transitive fan-in based algorithm is presented in column 5 of Table 5-1.

The benchmark circuits applied Dual-stage Pseudo PG with transitive fan-in based algorithm is simulated by Nanosim to evaluate the power consumption and the results are shown in Table 5-2.

The experiment results indicate that in addition to the first stage assignment, dual-stage structure reduces another 4.18% of power consumption. Our dual-stage Pseudo PG with the transitive fan-in based algorithm achieves a

Table 5-1 Results of dual stage with transitive fan-in based algorithm

Circuits	Controlled gates *Ns* / covered gates *N1*	Secondly controlled gate *N2*	*N2/Ns*	Run time(s)
C432	34/4	12	35.29%	0.60
C499	40/0	16	40.00%	0.40
C880	121/33	54	44.63%	0.60
C1355	16/0	4	25.00%	0.50
C1908	115/45	61	53.04%	0.50
C2670	303/31	73	24.09%	3.60
C3540	473/108	205	43.34%	9.70
C5315	887/142	277	31.23%	44.30
Avg.			37.08%	

Table 5-2 Power reduction of dual stage with transitive fan-in based algorithm

Circuits	Initial power *Pc*(μW)	Dual stage *Ps*"(μW)	Red.
C432	156.59	111.68	28.68%
C499	221.42	204.45	7.66%
C880	253.89	171.40	32.49%
C1355	352.3	277.97	21.10%
C1908	553.72	426.71	22.94%
C2670	816.35	578.13	29.18%
C3540	924.74	728.65	21.23%
C5315	1734.33	1343.44	22.54%
Avg.			23.23%

total power reduction of 23.23% on average. In contrast, 12.77% average power reduction has been obtained by previous *SW*-first algorithm. Compared with previous research, a further 10.46% power reduction has been achieved by the new approach proposed in this paper, which is a very significant improvement for the field of gate level power optimization.

5.5.2 Propagation Delay, Leakage Power, Area Analysis

Although the maximum depth constraint is exerted to avoid extreme degradation of timing performance in Pseudo PG method, the increase of propagation time and area overhead are unavoidable. To check the timing performance and area variation of the circuits before and after applying the Pseudo PG method with proposed approaches, the propagation delay and area of the circuits is evaluated by simulating the netlist after layout. In addition, the leakage power is also evaluated

Table 5-3 Propagation delay of dual stage structure

Circuits	Initial delay *Dc*(ns)	Dual-stage *D3*(ns)	Red. *Dc-D3/ Dc*
C432	2.75	2.90	5.45%
C499	1.79	1.79	0%
C880	2.25	2.41	7.11%
C1355	1.95	2.07	6.15%
C1908	2.71	2.92	7.75%
C2670	3.09	3.18	2.91%
C3540	3.88	4.11	5.93%
C5315	3.45	3.69	6.96%
Avg.			5.28%

Table 5-3 shows the delay increase. The propagation delay increase of single-stage Pseudo PG with transitive fan-in based algorithm is ranging from 0% to 6.96% (3.45% on average). Meanwhile, increase formed by applying proposed dual-stage Pseudo PG with transitive fan-in based clustering algorithm is ranging from 0% to 7.75% (5.28% on average). The proposed controlling path based algorithm results in 6.40% (on average) of area overhead and dual structure has 10.89% of area overhead compare with original circuits. The delay overhead is 2.75%. Table 5-4 shows the area overhead. Area overhead is 10.89% compared with the previous method. These overheads are moderate and the proposed method is useful.

Table 5-4 Area variation of dual stage structure

Circuits	Initial Area ($\times 10^3 \mu m^2$)	Dual-stage ($\times 10^3 \mu m^2$)	Red.
C432	3.08	3.65	18.51%
C499	4.70	5.26	11.91%
C880	5.71	6.66	16.64%
C1355	7.38	7.48	1.36%
C1908	11.43	12.01	5.07%
C2670	16.24	17.13	5.48%
C3540	23.57	27.50	16.67%
C5315	36.60	43.55	18.99%
Avg.			10.87%

The leakage power is also measured for benchmark circuits and Table 5-5 shows the results on the benchmark circuits with VDEC Rohm 0.18μm process. Leakage power is decreased in most of the circuits after applying the proposed algorithm and dual structure by the transistor stacking effect. The effect on dynamic power is successfully shown but the effect on static power is not clear since the cells in the process has very small leak current. Leak current is also expected to be small because of the stacking effect by adding extra controls, but that need to be checked with finer process rules

Table 5-5 Leakage power in dual stage structure

Circuits	Initial Leak. Power (nW)	Dual-stage (nW)	Red.
C432	1.72	1.01	41.28%
C499	4.45	4.48	-0.67%
C880	0.50	0.60	-20.00%
C1355	4.82	3.71	23.03%
C1908	1.93	1.68	12.95%
C2670	1.42	1.21	14.79%
C3540	3.17	1.83	42.27%
C5315	120.62	126.02	-4.48%
Avg.			13.65%

5.6 Summary

In this chapter, dual-stage structure for pseudo power gating (Pseudo PG) has been proposed to further reduce power consumption of the power-controlled block clustered by single-stage Pseudo PG. The concept of Dual-stage structure is that to adopt another Pseudo PG to the inside of

power-controlled block to stop unnecessary computation while the single-stage control signal takes non-controlling value.

To make the second stage realization fast and decrease the complexity of the program, Pseudo PG power-controlled gate related to a CV-propagation path is divided two different types: one is the gates generating side input of the starting gate on the CV-propagation path, and the other one is the gates computing the side inputs of other gates of the path. For the first type, another Pseudo PG is applied at second stage, but the second type can be controlled by signals on the CV-propagation path. By doing that we can realize the second stage with almost same complexity with single stage.

As a result, on averagely 37% of gates located in first stage power-controlled block obtained the opportunity to be controlled again.

The proposed dual-stage Pseudo PG structure is further implemented and 23.23% of average power reduction has been obtained with acceptable propagation delay. Compared with the method in Chapter 4, a 4.18% more power reduction has been achieved by the new approach proposed in this chapter, which with 5.28% delay overhead and 10.87% area overhead.

Chapter 6

Conclusions and Future Work

Power estimation and reduction have become primary concern for hardware design in recent years. Various researches have been done for reducing power consumption from different perspectives. The stage for investigation goes into aggressive power reduction techniques at every level of design abstraction. This dissertation contributes the dynamic power reduction method at gate level.

Through Chapter 1, power management technique are shown for low power CMOS technology in each abstraction level of design. We introduced the major components of power: dynamic, leakage, and short-circuit, followed by basic techniques to reduce these power components. Because of performance requirement, the switching speed is scaled up and that results in increase of dynamic power which depends on supply voltage, load capacitance, and switching activity. Low power design thus becomes the task of minimizing those parameters while retaining the required functionality.

Main part of power consumption is the switching activity of logic gates in the circuit. In the power reduction at gate level, switching activity of a circuit is the prime factor for optimization.

This dissertation discussed how to reduce switching activity at logic level. The proposed method is based on a controlling value of logic gate and on the power gating techniques. The method is called pseudo power gating, where a signal taking a controlling value is added to gate block as a control signal. The method is effective not only to the dynamic power reduction but also to the static power reduction.

6.1 Pseudo Power Gating Method and Switching Activity First Clustering

A controlling value of a gate decides its output with only one input. During the time of one input of a logic gate takes controlling value, other input values are not necessary and activities to decide other inputs are obsolete. Such unnecessary switching activities should be avoided. In reality, NAND/AND/ OR/NOR logic gates, which are widely used logic elements and the base of almost all logic circuits, have controlling value and unnecessary input change making the waste of power at running time. In an entire circuit, this accounts for a significant amount of overall power.

In this dissertation, a fine grained pseudo power gating (Pseudo PG) method based on the controlling value of a logic gate is proposed to avoid such wastage of energy on computing unnecessary input values. The Pseudo PG method adopts one of the input signals of a logic gate as a sleep signal to control the power of logic blocks that only involved in generating the other inputs of this gate. Output change of the controlled gates are inhibited and the power consumption can be reduced if the signal takes the controlling value. The signal can be used to control a power switch transistor for real power gating, but is used for reducing dynamic power. On the other hand, if the control signal takes

not-controlling value, power-controlled block work normally and the final output is decided by all inputs.

Distinguished from other power gating methods, the proposed power reduction method can reduce power consumption efficiently by optimizing the switching activity of combinational circuit. Pseudo PG method can be seen as a kind of signal gating, but is much more fine-grained in area and in time-direction and has smaller overhead compared with conventional signal gating methods.

The proposed Pseudo PG method might increase the depth of the critical path by adding an extra control signal. To cope with this issue, the steady maximum depth constraint is introduced to prevent the significant performance degradation. Related to the delay, it is observed that considerable glitches might happen when adding an extra control single to some gate. The solution is to remove control signal addition from the gate if the propagation path from this gate to the reconverged gate has odd number of inverting gates such as NOT, NAND, etc. This glitch reduction is demonstrated in the real power estimation.

An important aspect of Pseudo PG is to identify which gate is controlled by which signal. This information is required in advance to take proper gating decisions.

For a two-input gate having controlling value, one input can control a block related to the other input. However, both input cannot control other block at same time for the correctness of the circuit. This gives rise to an important problem with the Pseudo PG approach: How to select the control signal to obtain maximum power reduction? To cope with this problem, optimum clustering algorithm which can provide a good selection of control signal is desired to meets switching activity optimization demand. At first, the number of gates in a power-controlled block is focused and N-based algorithm is proposed, where N is the number of power-controlled gates. For each control signal candidates, we can

find a cluster of gates controlled by the signal. Note that the block has these block might not be controlled at same time. The maximum one is selected first and second largest one is selected next.

The candidate control signals are sorted by the number of controllable gates and one with the maximum number is selected as a control signal. Then the affected controlled block are recalculated and sorted again, and the same selection and recalculation procedure are applied until no new signal can be selected. N-based algorithm is implemented and applied to the benchmark circuit automatically and the switching activity reduction is confirmed.

However, for some blocks, it is observed that the switching activity of a controlled gate might be larger after a control signal is added. So switching activity-first clustering algorithm is devised to obtain the further switching activity reduction.

The objective is the switching activity reduction, so the reduction by adding a control signal is measured for each gate in the block. If a control signal inserted to a power-controlled gate, the switching activity of this gate might be reduced and the switching activities of cascaded fanout gates are changed until reaching the reconvergent gate. After the reconvergent gate, the switching activity is not affected due to the unchanging function. Therefore, evaluating the switching activity variation of the affected gates is important for the accurate estimation of the reduction.

By considering above affected reduction estimation, more accurate measurement can be obtained to represent the power saving. Thus, the switching activity (SW) first clustering algorithm is proposed by considering this affected reduction. In this algorithm, the candidate control signals are sorted with the reduction of switching activity and one with maximum value is selected as control signal and so on. As a result, averagely 12.77% of power saving is

obtained by applying the *SW*-first algorithm while fulfilling the requirement of maximum depth constraint.

6.2 Transitive Fan-in based Algorithm and Dual Stage Structure

Although, switching activity reduction in each power-controlled signal is correctly measured but several gates might lose the opportunity to be controlled due to the steady maximum depth constraint, and potential power reduction might be lost.

In order to cope with the steady maximum depth constraint to obtain further power savings, a transitive fan-in based clustering algorithm is proposed. In this algorithm, not only the direct input of a gate but also the transitive fan-ins are considered as a control signal candidate if the fan-in's controlling value becomes the controlling value of the focusing gate. Transitive fan-ins has the smaller depth and can become the control signal of the blocks of the other inputs of the focusing gate.

Another point improved in this algorithm is that the control signal is not inserted to a gate when both of input blocks are controlled already by the same control signal. Both inputs are controlled already to inhibit input change, so we need not to add the signal. The overhead by adding a control signal can be reduced by this control signal and power can be reduced.

Number of transistors is also considered in the estimation on the proposal. Since the power saving is estimated in terms of the amount of switching activity reduction, the switching activity of each gate should be represented at transistor level for high reliability. The number of transistors in each gate is different, so the amount of switching activities of entire circuit need to be

accumulated in proportion to the number of transistors in individual gates. For instance, if the number of transistors of NAND is normalized as 1.0, then that of AND gate is 1.5 since AND is constructed from NAND and Inverter. This coefficient can be used to measure switching activities.

By these proposals, averagely 6% of power is saved further than previous SW-first algorithm under the maximum depth constraint. The rates of power-controlled gates range from 2.71% to 35.40% (20.61% on average) of total gates in the circuit. The considerable reason is that the controllable gates excluded by the depth constraint in *SW*-first algorithm can be controlled with the smaller depth signal in transitive fan-in based algorithm.

In Pseudo PG, the controlled block does not work when the control signals take the controlling values, but the control block works when the control signals take non-controlling values. At that time, there might exist unnecessary switching activity in the controlled blocks.

To gate the unnecessary switching activity more radically, multi-stage structure is introduced. The concept of multi-stage is that to apply the Pseudo PG in the power-controlled block iteratively. Since the multiple stages higher than 2 cost too much, dual-stage structure is accepted in Pseudo PG.

When applying the second stage Pseudo PG, there are two different types in power-controlled gate related to a CV-propagation path: one is the gates generating side input of the starting gate on the CV-propagation path, and the other one is the gates computing the side inputs of other gates of the path. For the first type, another Pseudo PG is applied at second stage, but the second type can be controlled by signals on the CV-propagation path.

The proposed dual-stage Pseudo PG structure is further implemented and 23.23% of average power reduction with respect to the original circuit has been obtained with 5.28% delay overhead and 10.87% area overhead.

Compared with the transitive fan-in based single stage Pseudo PG, 4.18% more power reduction has been achieved.

6.3 Future Work

To reduce dynamic power, pseudo power gating method is introduced and several heuristic control signal selection/controlled gates clustering algorithms have been developed.

For the future work, we would like to implement the proposed pseudo power gating method to large scaled circuits. We should divide the circuit into several parts. In the case, the result is not optimum, but the power might be reduced using the local controlling relation in the divided part.

The effectiveness of the proposed method on dynamic power have been shown on benchmark circuit using VDEC Rohm 0.18μm process. The effect on dynamic power is successfully shown but the effect on static power is not clear since the cells in the process has very small leak current. Leak current is also expected to be small because of the stacking effect by adding extra controls, but that need to be checked with finer process rules.

The proposed pseudo power gating is for combinational circuit, but if we think about the sequential circuit, the proposed method is also applicable to optimize the power of combinational part in the circuit. In recent years, clock gating method is widely used to reduce the dynamic power consumption of registers by switching off unnecessary clock signals to the registers selectively. In the future, the combination of the proposed Pseudo PG and a clock gating method or a power gating method might be studied.

Acknowledgement

Completing my Dr. Eng. degree is the most challenging activity of my first 27 years of my life. It has been a great privilege to spend several years in the Graduate School of Information, Production and Systems, Waseda University and its members will always remain dear to me. It would not have been possible to write this doctoral dissertation without the help and support of the kind people around me.

My first debt of gratitude must go to my supervisor, Professor Shinji Kimura. He patiently provided the vision, encouragement and advice necessary for me to proceed through the academic program in the low power LSI design area. His prudential, profound view points and extraordinary motivation have been enlightening me in many ways. His gentle personality and rigorous attitude toward research will benefit my career as well as my personal life.

I also would like to express my sincere gratitude to Professor Takahiro Watanabe and Professor Takeshi Yoshimura, for their valuable suggestions on my research. Their thought provoking viewpoints and suggestion are very helpful to this dissertation. I deeply acknowledge for their patience and kind support. I will never forget their help and direction.

Meanwhile, I cannot forget the finance support and help from Professor Satoshi Goto. The experiment of this research cannot be completed without VLSI

design information and CAD software provided by VDEC (VLSI Design and Education Center in Tokyo University).

I also want to extend my gratitude to graduating seniors Dr. Chen Lei and Dr. Man Xin for their kind help and support. Lab members, Mr. Cheng Jun, Mr. Wang Bin and Mr. Wu Yifan, they are deserved my heartfelt appreciation for their support, guidance and helpful suggestions.

Finally, I will thank my parents and my friends for their encouragements and support in various ways for so long time. Without their great effort, current research achievements cannot be harvested.

Bibliography

[1] M. Keating, D. Flynn, R. Aitken, A. Gibbons, and D. Shi, Low Power Methodology Manual, pp.121-136, Springer, 2007.

[2] P. Panda, A. Shrivastava, B. Silpa and K. Gummidipudi, Power-efficient System Design, Springer, 2010.

[3] J. Rabaey, Low Power Design Essentials, Springer 2009.

[4] "International technology road map for semiconductors 2012 edition progress integration, device, and structures." [Online]. Available: http://www.itrs.net/Links/2012ITRS/Home2012.htm

[5] R. Menon, S. Chennupati, N. Samala, D. Radhakrishnan, and B. Izadi, "Power Optimized Combinational Logic Design," Proc. of International Conference on Embedded Systems and Applications, pp. 223–227, June 2003.

[6] C. Sharma "Low Power at Different Levels of VLSI Design and Clock Distribution Schemes," International Journal of Computer Technology and Applications, Vol. 2, No. 1, pp. 88-93, Jan. 2011.

[7] B. Amelifard, F. Fallah, M. Pedram, "Low-power Fanout Optimization Using MTCMOS and Multi-Vt Techniques," Proc. of the 2006 International Symposium on Low Power Electronics and Design (ISLPED'06), pp. 334-337, Oct. 2006.

[8] C. Chen, A. Srivastava, and M. Sarrafzadeh, "On Gate Level Power Optimization Using Dual-Supply Voltages," IEEE Trans. VLSI Systems, Vol. 9, No. 5, pp. 616-629, Oct. 2001.

[9] T. Yamamoto, K. Hironaka, Y. Hayakawa, M. Kimura, H. Amano, and K.

Usami, “Dynamic VDD Switching Technique and Mapping Optimization in Dynamically Reconfigurable Processor for Efficient Energy Reduction,” Proc. of the 7th International Symposium on Applied Reconfigurable Computing, pp. 230-241, Mar. 2011.

[10] R. Menon, S. Chennupati, N. Samala, D. Radhakrisnan, and B. Izadi, “Switching Activity Minimization in Combinational Logic Design,” Proc. International Conference on Embedded Systems and Applications, pp.47-53, June 2004.

[11] L. Li, K. Choi, S. Park, M. Chung, “Novel RT level methodology for low power by using wasting toggle rate based clock gating,” SoC Design Conference, pp. 484-487, Nov. 2009.

[12] N. Banerjee, A. Raychowdhury, K. Roy, S. Bhunia, and. H. Mahmoodi, “Novel Low-Overhead Operand Isolation Techniques for Low-Power Datapath Synthesis,” Proc. International Conference on Computer Design, pp.1034-1039, Sep. 2006.

[13] N. Honarmand and A. Afzali-K Low Power Combinational Multiplier Using Data driven Signal Gating,” Proc. IEEE Asia Pacific Conference on Circuits and Systems, pp. 1456-1459, Dec 2006.

[14] L. Chen, T. Horiyama, Y. Nakamura, and S. Kimura, “Fine-grained power gating based on the controlling value of logic elements,” IEICE Trans. Fundamentals, Vol. E91-A, No.12, pp. 3531-3538, Dec. 2008.

[15] L. Chen, and S. Kimura, “Optimizing Controlling Value-Based Power Gating with Gate Count and Switching Activity,” IEICE Trans. Fundamentals, Vol.E92-A, No.12, pp. 3111-3118, Dec. 2009.

[16] Y. Jin, and S. Kimura, “On Gate Level Power Optimization of Combinational Circuits Using Pseudo Power Gating,” IEICE Trans. Fundamentals, Vol. E95-A, No.12, pp. 2191-2198, Dec. 2012.

[17] Y. JIN, Z. DU and S. KIMURA, “Dual-Stage Pseudo Power Gating with Advanced Clustering Algorithm for Gate Level Power Optimization,” IEICE Trans. Fundamentals, Vol. E96-A, No.12, 8 pages, Dec. 2013. (to appear)

[18] S.B. Akers. “Binary Decision Diagrams.” IEEE Trans. On Comp., Vol. 27, No. 7, pp. 509-516, June 1978.

[19] T. Hashida and K. Usami, "Power-Switch Clustering Method for Static Timing Analysis," Proc. of the 23rd International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC'08), pp. 217-220, Jul. 2008.

[20] C. Hwang, P. Rong and M. Pedram, "Sleep Transistor Distribution in Row-Based MTCMOS Designs," Proc. of the 17th ACM Great Lakes symposium on VLSI (GLSVLSI), pp. 235-240, 2007.

[21] S. Nakamura, I. Taniguchi, H. Tomiyama and M. Fukui, "A Basic-Block Level Optimistic Energy Estimation for Power-Gated VLIW Data-Path Model," Proc. of SASIMI 2013, pp.354-359, Oct. 2013.

[22] S. Kim, S. Kosonocky, D. Knebeland K. Stawiasz, "A Multi-Mode Power Gating Structure for Low-Voltage Deep-Submicron CMOS ICs," IEEE Transaction on Circuits and Systems II: Express Briefs, Vol. 54, No. 7, pp.586-590, July 2007

[23] A. Kahng, S, Kang and B. Park, "Active-Mode Leakage Reduction with Data-Retained Power Gating," Design, Automation & Test in Europe Conference & Exhibition, pp.1209-1214, Mar. 2013.

[24] D. Flynn, "Power gating applied to MP-SoCs for standby-mode power management", Proc. of the 50th ACM/EDAC/IEEE Design Automation Conference (DAC 2013), pp.1-5, May 2013.

[25] R. Puri, "Minimizing power under performance constriant," Proc. of International Conference on Integrated Circuit Design and Technology (ICICDT'04), pp. 159-163, 2004.

[26] H. Mangassarian, A. Veneris, and F. Najm, “Maximum Circuit Activity Estimation Using Pseudo-Boolean Satisfiability,” IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, Vol. 31, No. 2, pp. 271-284, Feb. 2012.

[27] M. Hosseinabady, P. Lotfi-Kamran, J. Mathew, S. Mohanty, and D. Pradhan, “Single-Event Transient Analysis in High Speed Circuits,” 2011 International Symposium on Electronic System Design, pp. 19-21, Dec. 2011.

[28] Y. Chen, Y. Shi, K. Lai, G. Hui and S. Chang, "Efficient Multiple-Bit Retention Register Assignment for Power Gated Design: Concept and Algorithms," Proc.

of 2012 IEEE/ACM International Conference on Computer-Aided Design (ICCAD 2012), pp.309-316, Nov. 2012.

[29] L. Zhang, B. Tiwana, Z. Qian, Z. Wang, R. Dick, Z. Mao, and L. Yang, "Accurate online power estimation and automatic battery behavior based power model generation for smartphones" Proc. of IEEE/ACM/IFIP Int. Conf. Hardware/Software Codesign and System Synthesis, pp. 105-114, 2010.

[30] A. Srivastava, D. Sylvester, and D. Blaauw, “Power minimization using simultaneous gate sizing, dual-vdd and dual-vth assignment,” Proc. of the Design Automation Conference, pp. 783–787, 2004.

[31] Xin MAN, Takashi HORIYAMA and Shinji KIMURA, “Automatic Multi-Stage Clock Gating Optimization Using ILP Formulation”, IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences, Vol. E95-A, No. 8, pp. 1347-1358, Aug. 2012.

[32] Xin MAN, Takashi HORIYAMA and Shinji KIMURA, “Power Optimization of Sequential Circuits Using Switching Activity Based Clock Gating”, IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences, Vol. E93-A, No. 12, pp. 2472-2480, Dec. 2010.

[33] C. Chen, K. Bowman, C. Augustine and Z. Zhang, "Minimum Supply Voltage for Sequential Logic Circuits in a 22nm Technology," Proc. of IEEE International Symposium on Low Power Electronics and Design (ISLPED 2013), pp. 181-186, Sep. 2013.

[34] K. Kim, H. Nan and K. Choi, "Ultralow-Voltage Power Gating Structure Using Low Threshold Voltage," IEEE Trans. on circuits and systems-II: Express Briefs, Vol. 56, No. 12, pp.926-930, Dec. 2009,

[35] Y. Lee, D. Jeong and T. Kim, "Simultaneous Control of Power/Ground Current, Wakeup Time and Transistor Overhead in Power Gated Circuits," Proc. of IEEE/ACM International Conference on Computer-Aided Design (ICCAD 2008), pp. 169-172, Nov. 2008.

[36] P. Lindgren, M. Kerttu, M. Thornton and R. Drechsler, "Low Power Optimization Technique for BDD Mapped Circuits," Proc. of Asia and South Pacific Design Automation Conference (ASP-DAC 2001), pp. 615-621, Jan. 2001.

[37] S. Yang, W. Wolf, N. Vijaykrishnan, Y. Xie, W. Wang, "Accurate Stacking Effect Macro-modeling of Leakage Power in Sub-100nm Circuits," Proc. of the 18th International Conference on VLSI Design held jointly with 4th International Conference on Embedded Systems Design (VLSID'05), pp. 165-170, Jan. 2005.

[38] M. Munch, B. Wurth, R. Mehra and J. Sproch, "Automating RT-level operand isolation to minimize power consumption in datapaths," Proc. of Design, Automation and Test in Europe Conference and Exhibition 2000, pp.624-631, Mar.2000.

List of Publications

Journal Papers

[1] Yu JIN, Zhe DU and Shinji KIMURA, “Dual-Stage Pseudo Power Gating with Advanced Clustering Algorithm for Gate Level Power Optimization,” IEICE Trans. Fundamentals, Vol.E96-A, No.12, pp.2568-2575, Dec. 2013.

[2] Yu JIN a tion of Combinational Circuits Using Pseudo Power Gating,” IEICE Trans. Fundamentals, Vol.E95-A, No.12, pp. 2191-2198, Dec. 2012.

International conference papers

[3] Yu JIN, Zhe DU and Shinji KIMURA, “An Advanced Pseudo Power Gating for Gate Level Power Optimization,” Proc. of ITC-CSCC 2013, Yeosu, Korea, pp.270-273, June 2013.

[4] Yu JIN a -Stage Power Gating Based on Controlling Values of Logic Gates,” Proc. The IEEE 9th International Conference on ASIC, pp.87-90, Oct. 2011.